Open Source Used In
security_automation identity-guest-cs

Cisco Systems, Inc.
www.cisco.com

Cisco has more than 200 offices worldwide. Addresses, phone numbers, and fax numbers are listed on the Cisco website at www.cisco.com/go/offices.
This document contains licenses and notices for open source software used in this product. With respect to the free/open source software listed in this document, if you have any questions or wish to receive a copy of any source code to which you may be entitled under the applicable free/open source license(s) (such as the GNU Lesser/General Public License), please contact us at external-opensource-requests@cisco.com.

In your requests please include the following reference number 78EE117C99-1256001742

## Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Package Name</th>
<th>Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>avro</td>
<td>1.8.2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1.1.1</td>
</tr>
<tr>
<td>1.2</td>
<td>java-uuid-generator</td>
<td>4.0.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1.2.1</td>
</tr>
<tr>
<td>1.3</td>
<td>xml-apis</td>
<td>2.0.2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1.3.1</td>
</tr>
<tr>
<td>1.4</td>
<td>netty-codec-http</td>
<td>4.1.69.Final</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1.4.1</td>
</tr>
<tr>
<td>1.5</td>
<td>spring-cloud-consul-core</td>
<td>3.0.4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1.5.1</td>
</tr>
<tr>
<td>1.6</td>
<td>async-http-client</td>
<td>2.11.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1.6.1</td>
</tr>
<tr>
<td>1.7</td>
<td>postgresql-jdbc</td>
<td>42.2.24</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1.7.1</td>
</tr>
<tr>
<td>1.8</td>
<td>spring-beans</td>
<td>5.3.12</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1.8.1</td>
</tr>
<tr>
<td>1.9</td>
<td>fast-infoset</td>
<td>1.2.13</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1.9.1</td>
</tr>
<tr>
<td>1.10</td>
<td>swagger-jersey-jaxrs</td>
<td>1.5.21</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1.10.1</td>
</tr>
<tr>
<td>1.11</td>
<td>json-java</td>
<td>20200518</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1.11.1</td>
</tr>
<tr>
<td>1.12</td>
<td>zookeeper-jute</td>
<td>3.5.5</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1.12.1</td>
</tr>
<tr>
<td>1.13</td>
<td>netty-codec-mqtt</td>
<td>4.1.69.Final</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1.13.1</td>
</tr>
</tbody>
</table>
1.14 concurrentlinkedhashmap-lru 1.3.2
   1.14.1 Available under license
1.15 jackson-datatype-jsr310 2.13.0
   1.15.1 Available under license
1.16 javax-annotation-api 1.3.2
   1.16.1 Available under license
1.17 hikaricp 3.4.5
   1.17.1 Available under license
1.18 avro 1.10.0
   1.18.1 Available under license
1.19 async-http-client-netty-utils 2.11.0
   1.19.1 Available under license
1.20 jackson-module-parameter-names 2.13.0
   1.20.1 Available under license
1.21 kafka-schema-registry-client 5.3.0
   1.21.1 Available under license
1.22 hibernate-validator 6.1.7.Final
   1.22.1 Available under license
1.23 jackson-databind 2.13.0
   1.23.1 Available under license
1.24 lang-tag 1.4.4
   1.24.1 Available under license
1.25 config 1.3.4
   1.25.1 Available under license
1.26 versly-wsdoc 1.1.010
1.27 reactive-streams 1.0.3
   1.27.1 Available under license
1.28 json-path 2.4.0
   1.28.1 Available under license
1.29 commons-collections 3.2.1
   1.29.1 Available under license
1.30 jersey-servlet 1.19.1
1.31 framework-tool-plugin-doc 2.1.25-SNAPSHOT
   1.31.1 Available under license
1.32 dom4j 2.1.3
   1.32.1 Available under license
1.33 native-protocol 1.4.12
   1.33.1 Available under license
1.34 common-utils 5.3.0
1.34.1 Available under license
1.35 commons-text 1.9
1.35.1 Available under license
1.36 jnr-a64asm 1.0.0
1.36.1 Available under license
1.37 st4 4.0.8
1.37.1 Available under license
1.38 jcip-annotation 1.0.1
1.38.1 Available under license
1.39 logback-core 1.2.6
1.39.1 Available under license
1.40 netty-resolver 4.1.69.Final
1.40.1 Available under license
1.41 failureaccess 1.0.1
1.42 httpcomponents-mime 4.5.13
1.42.1 Available under license
1.43 jnr-ffi 2.2.0
1.43.1 Available under license
1.44 netty-codec 4.1.69.Final
1.44.1 Available under license
1.45 jsr311-api 1.1.1
1.45.1 Available under license
1.46 velocity 1.7
1.46.1 Available under license
1.47 slf4j-log4j 1.7.32
1.47.1 Available under license
1.48 jnr-constants 0.10.0
1.48.1 Available under license
1.49 server-sdk 6.0.1.0
1.49.1 Available under license
1.50 jackson 1.9.2
1.50.1 Available under license
1.51 jcommander 1.78
1.51.1 Available under license
1.52 jctools-core 3.1.0
1.52.1 Available under license
1.53 httpclient-cache 4.5.13
1.53.1 Available under license
1.54 spring-rabbit 2.3.11
1.76 radeox-oro 0.9  
   1.76.1 Available under license

1.77 guava 30.1.1-jre  
   1.77.1 Available under license

1.78 jetty-continuation 9.4.44.v20210927  
   1.78.1 Available under license

1.79 mail 1.4.4  
   1.79.1 Available under license

1.80 jsoup 1.11.3  
   1.80.1 Available under license

1.81 jackson-dataformat-yaml 2.13.0  
   1.81.1 Available under license

1.82 jackson-xc 1.9.2  
   1.82.1 Available under license

1.83 classmate 1.5.1  
   1.83.1 Available under license

1.84 json-smart 2.3  
   1.84.1 Available under license

1.85 jmespath-java 1.11.868  
   1.85.1 Available under license

1.86 metrics-httpclient 4.1.26  
   1.86.1 Available under license

1.87 jain-sip-ri 1.2.148.3  
   1.87.1 Available under license

1.88 protobuf-java 2.2.0
1.89 jetty-util 9.4.44.v20210927  
   1.89.1 Available under license

1.90 qdox 1.12.1
1.91 jaxb-api 2.3.0  
   1.91.1 Available under license

1.92 libffi 1.2  
   1.92.1 Available under license

1.93 dom4j 1.6.1  
   1.93.1 Available under license

1.94 jackson-datatype-guava 2.13.0  
   1.94.1 Available under license

1.95 guava 24.1.1-jre
1.96 jackson-annotations 2.13.0  
   1.96.1 Available under license
<table>
<thead>
<tr>
<th>Version</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.97</td>
<td>jetty-io 9.4.44.v20210927</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.98</td>
<td>framework-supports 2.1.25-SNAPSHOT</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.99</td>
<td>oauth2-oidc-sdk 8.36.2</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.100</td>
<td>okio 1.17.2</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.101</td>
<td>json-schema-validator 1.0.57</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.102</td>
<td>jackson-datatype-jdk8 2.13.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.103</td>
<td>objenesis 3.1</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.104</td>
<td>activation 1.1</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.105</td>
<td>spring-cloud-commons 3.0.4</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.106</td>
<td>msal4j 1.2.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.107</td>
<td>quality-check 1.3</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.108</td>
<td>freemarker 2.3.31</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.109</td>
<td>uadetector-resources 2014.10</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.110</td>
<td>jackson-jaxrs 1.9.2</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.111</td>
<td>commons-codec 1.15</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.112</td>
<td>metrics 4.1.26</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.113</td>
<td>commons-compress 1.18</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.114</td>
<td>joda-time 2.10.6</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.115</td>
<td>hibernate-validator 5.0.2.Final</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.116</td>
<td>javax-activation 1.2.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.117</td>
<td>swagger-annotations 1.5.21</td>
<td>Available under license</td>
</tr>
</tbody>
</table>
1.117.1 Available under license
1.118 **accessors-smart 1.2**
   1.118.1 Available under license
1.119 **dnsjava 2.1.7**
   1.119.1 Available under license
1.120 **jacoco-core 0.8.5**
   1.120.1 Available under license
1.121 **commons-configuration 1.10**
   1.121.1 Available under license
1.122 **asm-analysis 7.1**
   1.122.1 Available under license
1.123 **jedis 3.3.0**
   1.123.1 Available under license
1.124 **jackson-mapper-asl 1.9.2**
   1.124.1 Available under license
1.125 **log4j-api 2.17.0**
   1.125.1 Available under license
1.126 **httpcomponents-client 4.5.13**
   1.126.1 Available under license
1.127 **jffi 1.3.0**
1.128 **uadetector-core 0.9.22**
   1.128.1 Available under license
1.129 **spring-cloud-starter-consul-config 3.0.4**
   1.129.1 Available under license
1.130 **jersey-server 1.19.1**
1.131 **classgraph 4.8.21**
   1.131.1 Available under license
1.132 **common-config 5.3.0**
   1.132.1 Available under license
1.133 **j2objc-annotations 1.3**
1.134 **log4j-over-slf4j 1.7.32**
   1.134.1 Available under license
1.135 **netty-transport-native-epoll 4.1.69.Final**
   1.135.1 Available under license
1.136 **commons-lang3 3.11**
   1.136.1 Available under license
1.137 **netty-transport-native- unix-common 4.1.69.Final**
   1.137.1 Available under license
1.138 **netty-transport-native-kqueue 4.1.69.Final**
1.138.1 Available under license

1.139 **hystrix-core** 1.5.18
1.139.1 Available under license

1.140 **aws-java-sdk-cognitoidp** 1.11.868
1.140.1 Available under license

1.141 **javassist** 3.27.0-GA
1.141.1 Available under license

1.142 **consul-api** 1.4.5
1.142.1 Available under license

1.143 **netty-buffer** 4.1.69.Final
1.143.1 Available under license

1.144 **jakarta-annotation-api** 1.3.5
1.144.1 Available under license

1.145 **esri-geometry-api** 1.2.1
1.145.1 Available under license

1.146 **content-type** 2.1
1.146.1 Available under license

1.147 **snappy-java** 1.1.7.7
1.147.1 Available under license

1.148 **jackson-datatype-joda** 2.13.0
1.148.1 Available under license

1.149 **commons-logging** 1.2
1.149.1 Available under license

1.150 **activation** 1.2.0
1.150.1 Available under license

1.151 **zstd** 1.4.4
1.151.1 Available under license

1.152 **spring-cloud-consul-config** 3.0.4
1.152.1 Available under license

1.153 **aws-java-sdk-s3** 1.11.415
1.153.1 Available under license

1.154 **bean-validation-api** 2.0.1.Final
1.154.1 Available under license

1.155 **msgpack-core** 0.8.20
1.155.1 Available under license

1.156 **influxdb-java** 2.20
1.156.1 Available under license

1.157 **netty-transport-udt** 4.1.69.Final
1.157.1 Available under license
1.158 scim2-sdk-common 2.1.1
   1.158.1 Available under license
1.159 hibernate 5.4.32.Final
   1.159.1 Available under license
1.160 spring-aspects 5.3.12
   1.160.1 Available under license
1.161 mime-pull 1.9.15
   1.161.1 Available under license
1.162 jetty-servlets 9.4.44.v20210927
   1.162.1 Available under license
1.163 curator-client 5.1.0
   1.163.1 Available under license
1.164 log4j-to-slf4j 2.17.0
   1.164.1 Available under license
1.165 commons-beanutils 1.8.3
   1.165.1 Available under license
1.166 java-driver-query-builder 4.10.0
   1.166.1 Available under license
1.167 jersey-json 1.19.1
1.168 jackson-dataformat-cbor 2.13.0
   1.168.1 Available under license
1.169 paranamer 2.7
1.170 jboss-transaction-api 1.1.1.Final
   1.170.1 Available under license
1.171 jersey 1.19.1
   1.171.1 Available under license
1.172 netty-transport 4.1.69.Final
   1.172.1 Available under license
1.173 netty-resolver-dns-native-macos 4.1.69.Final
   1.173.1 Available under license
1.174 gson 2.8.8
   1.174.1 Available under license
1.175 rxjava 1.3.8
   1.175.1 Available under license
1.176 spring-jdbc 5.3.12
   1.176.1 Available under license
1.177 activation-api 1.2.0
   1.177.1 Available under license
1.178 spring-retry 1.3.1
1.178.1 Available under license
1.179 antlr 2.7.7
  1.179.1 Available under license
1.180 httpcomponents-core 4.4.14
  1.180.1 Available under license
1.181 args4j 2.0.29
1.182 netty-reactive-streams 2.0.4
  1.182.1 Available under license
1.183 jsr305 3.0.2
  1.183.1 Available under license
1.184 kotlin 1.4.32
  1.184.1 Available under license
1.185 listenablefuture 1.0
  1.185.1 Available under license
1.186 cloning 1.9.0
  1.186.1 Available under license
1.187 lombok 1.18.16
  1.187.1 Available under license
1.188 spring-integration-core 5.4.11
  1.188.1 Available under license
1.189 istack-commons 3.0.5
  1.189.1 Available under license
1.190 jnr-posix 3.1.2
  1.190.1 Available under license
1.191 unboundid-ldapsdk-commercial-edition 3.2.1
  1.191.1 Available under license
1.192 retrofit 2.9.0
  1.192.1 Available under license
1.193 snake-yaml 1.27
  1.193.1 Available under license
1.194 asm 7.1
  1.194.1 Available under license
1.195 jandex 2.2.3.Final
  1.195.1 Available under license
1.196 commons-io 2.11.0
  1.196.1 Available under license
1.197 spring-security-crypto 5.4.9
  1.197.1 Available under license
1.198 listenablefuture 9999.0-empty-to-avoid-conflict-with-guava
1.198.1 Available under license
1.199 Spring AMQP 2.3.11
  1.199.1 Available under license
1.200 Java Statsd Client 3.1.0
  1.200.1 Available under license
1.201 HttpClient 4.1.4
  1.201.1 Available under license
1.202 Commons Digest 1.8.1
  1.202.1 Available under license
1.203 Aalto XMl 1.0.0
  1.203.1 Available under license
1.204 Avro Guava Dependencies 1.8.2
  1.204.1 Available under license
1.205 Jaxinject 1
  1.205.1 Available under license
1.206 Abego Treelayout Core 1.0.1
  1.206.1 Available under license
1.207 Jcl Over SLF4j 1.7.32
  1.207.1 Available under license
1.208 Netty Handler Proxy 4.1.69.Final
  1.208.1 Available under license
1.209 Curator Recipes 5.1.0
  1.209.1 Available under license
1.210 Jetty HTTP 9.4.44.v20210927
  1.210.1 Available under license
1.211 Netty Codec Stomp 4.1.69.Final
  1.211.1 Available under license
1.212 Swagger 1.5.21
1.213 Netty Transport Sctp 4.1.69.Final
  1.213.1 Available under license
1.214 Spring TX 5.3.12
  1.214.1 Available under license
1.215 Jakarta Validation API 2.0.2
  1.215.1 Available under license
1.216 Spring AOP 5.3.12
  1.216.1 Available under license
1.217 Audience Annotations 0.5.0
  1.217.1 Available under license
1.218 Scala 2.13.1
1.218.1 Available under license
1.219 antlr 4.5
1.220 netty 4.1.69.Final
   1.220.1 Available under license
1.221 okhttp 3.14.9
   1.221.1 Available under license
1.222 txw2 2.3.5
   1.222.1 Available under license
1.223 rtx 2.1.7
   1.223.1 Available under license
1.224 antlr 3.5.2
   1.224.1 Available under license
1.225 spring-data-commons 2.4.14
   1.225.1 Available under license
1.226 jackson-xc 2.13.0
   1.226.1 Available under license
1.227 spring-context 5.3.12
   1.227.1 Available under license
1.228 jacoco-agent 0.8.5
   1.228.1 Available under license
1.229 jaxb-runtime 2.3.0
   1.229.1 Available under license
1.230 ion-java 1.0.2
   1.230.1 Available under license
1.231 jakarta-el 3.0.4
   1.231.1 Available under license
1.232 checker-qual 3.8.0
   1.232.1 Available under license
1.233 jackson-jaxrs-base 2.13.0
   1.233.1 Available under license
1.234 xz-java 1.5
   1.234.1 Available under license
1.235 postgresql 42.2.24
   1.235.1 Available under license
1.236 netty-codec-memcache 4.1.69.Final
   1.236.1 Available under license
1.237 jaxb-core 2.3.0
1.238 spring-integration-kafka 5.4.11
   1.238.1 Available under license
<table>
<thead>
<tr>
<th>Version</th>
<th>Package Name</th>
<th>Version Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.239</td>
<td>lettuce-core 6.1.2.RELEASE</td>
<td>1.239.1 Available under license</td>
</tr>
<tr>
<td>1.240</td>
<td>java-driver-shaded-guava 25.1-jre-graal-sub-1</td>
<td>1.240.1 Available under license</td>
</tr>
<tr>
<td>1.241</td>
<td>curator-framework 5.1.0</td>
<td>1.241.1 Available under license</td>
</tr>
<tr>
<td>1.242</td>
<td>commons-beanutils-core 1.8.3</td>
<td>1.242.1 Available under license</td>
</tr>
<tr>
<td>1.243</td>
<td>aspect-j 1.9.7</td>
<td>1.243.1 Available under license</td>
</tr>
<tr>
<td>1.244</td>
<td>byte-buddy 1.10.22</td>
<td>1.244.1 Available under license</td>
</tr>
<tr>
<td>1.245</td>
<td>netty-codec-socks 4.1.69.Final</td>
<td>1.245.1 Available under license</td>
</tr>
<tr>
<td>1.246</td>
<td>netty-codec-xml 4.1.69.Final</td>
<td>1.246.1 Available under license</td>
</tr>
<tr>
<td>1.247</td>
<td>spring-web 5.3.12</td>
<td>1.247.1 Available under license</td>
</tr>
<tr>
<td>1.248</td>
<td>spring-kafka 2.7.1</td>
<td>1.248.1 Available under license</td>
</tr>
<tr>
<td>1.249</td>
<td>jakarta xml bind api 2.3.3</td>
<td>1.249.1 Available under license</td>
</tr>
<tr>
<td>1.250</td>
<td>httpcore-nio 4.4.14</td>
<td>1.250.1 Available under license</td>
</tr>
<tr>
<td>1.251</td>
<td>jain-sip-api 1.2.1</td>
<td>1.251.1 Available under license</td>
</tr>
<tr>
<td>1.252</td>
<td>netty-handler 4.1.69.Final</td>
<td>1.252.1 Available under license</td>
</tr>
<tr>
<td>1.253</td>
<td>asm-util 7.1</td>
<td>1.253.1 Available under license</td>
</tr>
<tr>
<td>1.254</td>
<td>logging-interceptor 3.14.9</td>
<td>1.254.1 Available under license</td>
</tr>
<tr>
<td>1.255</td>
<td>jldap 2013.08.30.1433</td>
<td>1.255.1 Available under license</td>
</tr>
<tr>
<td>1.256</td>
<td>jul-to-slf4j 1.7.32</td>
<td>1.256.1 Available under license</td>
</tr>
<tr>
<td>1.257</td>
<td>jnr-x86asm 1.0.2</td>
<td>1.257.1 Available under license</td>
</tr>
<tr>
<td>1.258</td>
<td>spring-security-jwt 1.0.10.RELEASE</td>
<td></td>
</tr>
</tbody>
</table>
1.258.1 Available under license

**1.259 bouncy-castle 1.68**
1.259.1 Available under license

**1.260 nimbus-jose-jwt 8.20**
1.260.1 Available under license

**1.261 kafka-clients 2.7.1**
1.261.1 Available under license

**1.262 spring-javaconfig 1.0.0.m3**
1.262.1 Available under license

**1.263 aop-alliance 5.3.12**
1.263.1 Available under license

**1.264 zookeeper 3.5.5**
1.264.1 Available under license

**1.265 annotations 20.1.0**
1.265.1 Available under license

**1.266 hdrhistogram 2.1.12**
1.266.1 Available under license

**1.267 commons-beanutils 1.9.4**
1.267.1 Available under license

**1.268 commons-validator 1.6**
1.268.1 Available under license

**1.269 spring-security-rsa 1.0.10.RELEASE**
1.269.1 Available under license

**1.270 reflections 0.9.11**
1.270.1 Available under license

**1.271 aws-java-sdk-core 1.11.415**
1.271.1 Available under license

**1.272 jackson 2.13.0**
1.272.1 Available under license

**1.273 activation-api 1.2.2**
1.273.1 Available under license

**1.274 spring-cloud-context 3.0.4**
1.274.1 Available under license

**1.275 converter-moshi 2.9.0**
1.275.1 Available under license

**1.276 oro 2.0.7**
1.276.1 Available under license

**1.277 kafka-avro-serializer 5.3.0**
1.277.1 Available under license
<table>
<thead>
<tr>
<th>Version</th>
<th>Library/Module</th>
<th>Available under license</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.278</td>
<td>swagger-core 1.5.21</td>
<td></td>
</tr>
<tr>
<td>1.278.1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.279</td>
<td>javax-persistence-api 2.2</td>
<td></td>
</tr>
<tr>
<td>1.279.1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.280</td>
<td>slf4j 1.7.32</td>
<td></td>
</tr>
<tr>
<td>1.280.1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.281</td>
<td>netty-codec-smtp 4.1.69.Final</td>
<td></td>
</tr>
<tr>
<td>1.281.1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.282</td>
<td>org.jacoco.agent.rt 0.8.5</td>
<td></td>
</tr>
<tr>
<td>1.282.1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.283</td>
<td>janino 3.1.6</td>
<td></td>
</tr>
<tr>
<td>1.283.1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.284</td>
<td>spring-cloud-starter 3.0.4</td>
<td></td>
</tr>
<tr>
<td>1.284.1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.285</td>
<td>asm-tree 7.1</td>
<td></td>
</tr>
<tr>
<td>1.285.1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.286</td>
<td>netty-codec-redis 4.1.69.Final</td>
<td></td>
</tr>
<tr>
<td>1.286.1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.287</td>
<td>scim2-sdk-client 2.1.1</td>
<td></td>
</tr>
<tr>
<td>1.287.1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.288</td>
<td>spring-data-cassandra 3.1.14</td>
<td></td>
</tr>
<tr>
<td>1.288.1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.289</td>
<td>jettison 1.1</td>
<td></td>
</tr>
<tr>
<td>1.289.1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.290</td>
<td>moshi 1.8.0</td>
<td></td>
</tr>
<tr>
<td>1.290.1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.291</td>
<td>mixpanel-java 1.5.0</td>
<td></td>
</tr>
<tr>
<td>1.291.1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.292</td>
<td>stax 4.0.0</td>
<td></td>
</tr>
<tr>
<td>1.292.1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.293</td>
<td>barchart-udt-bundle 2.3.0</td>
<td></td>
</tr>
<tr>
<td>1.293.1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.294</td>
<td>jersey-spring 1.19.1</td>
<td></td>
</tr>
<tr>
<td>1.294.1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.295</td>
<td>swagger-jaxrs 1.5.21</td>
<td></td>
</tr>
<tr>
<td>1.295.1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.296</td>
<td>jboss-logging 3.4.2.Final</td>
<td></td>
</tr>
<tr>
<td>1.296.1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.297</td>
<td>spring-web-mvc 5.3.12</td>
<td></td>
</tr>
<tr>
<td>1.297.1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Version</td>
<td>Description</td>
<td>License</td>
</tr>
<tr>
<td>---------</td>
<td>-------------------------------------------------------------------------------</td>
<td>-----------</td>
</tr>
<tr>
<td>1.298</td>
<td>java-driver-metrics-micrometer 4.10.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.299</td>
<td>jackson-jaxrs 2.13.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.300</td>
<td>netty-codec-haproxy 4.1.69.Final</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.301</td>
<td>jaxen 1.2.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.302</td>
<td>metrics-influxdb 1.1.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.303</td>
<td>asm-commons 7.1</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.304</td>
<td>java-driver-core 4.10.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.305</td>
<td>spring-cloud-starter-consul 3.0.4</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.306</td>
<td>latencyutils 2.0.3</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.307</td>
<td>ojdbc 19.3.0.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.308</td>
<td>zstd-jni 1.4.5-6</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.309</td>
<td>commons-compiler 3.1.6</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.310</td>
<td>netty-codec-dns 4.1.69.Final</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.311</td>
<td>amqp-client 5.10.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.312</td>
<td>aws-java-sdk-kms 1.11.415</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.313</td>
<td>hibernate-commons-annotations 5.1.2.Final</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.314</td>
<td>spring-cloud-starter-bootstrap 3.0.4</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.315</td>
<td>spring-framework 5.3.12</td>
<td>Available under license</td>
</tr>
</tbody>
</table>

1.1 avro 1.8.2
1.1.1 Available under license:

Apache Avro
Copyright 2009-2017 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.2 java-uuid-generator 4.0.1

1.2.1 Available under license:

This copy of Java UUID Generator (JUG) library is licensed under Apache (Software) License,
version 2.0 ("the License").
See the License for details about distribution rights, and the specific rights regarding derivate works.

You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0
Java UUID generator library has been written by Tatu Saloranta (tatu.saloranta@iki.fi)

Other developers who have contributed code are:

* Eric Bie contributed extensive unit test suite which has helped ensure high implementation quality

1.3 xml-apis 2.0.2

1.3.1 Available under license:

No license file was found, but licenses were detected in source scan.

// NO WARRANTY! This class is in the Public Domain.
/**
 * Base class for deriving an XML filter.
 * *
 * <blockquote>
 * <em>This module, both source code and documentation, is in the
This class is designed to sit between an `XMLReader` and the client application's event handlers. By default, it does nothing but pass requests up to the reader and events on to the handlers unmodified, but subclasses can override specific methods to modify the event stream or the configuration requests as they pass through.

@since SAX 2.0
@author David Megginson,
@version 2.0
@see org.xml.sax.XMLFilter
@see org.xml.sax.XMLReader
@see org.xml.sax.EntityResolver
@see org.xml.sax.DTDHandler
@see org.xml.sax.ContentHandler
@see org.xml.sax.ErrorHandler
* to decide whether to throw an exception for different types of
* errors and warnings. Note, however, that there is no requirement that
* the parser continue to provide useful information after a call to
* {@link #fatalError fatalError} (in other words, a SAX driver class
* could catch an exception and report a fatalError).</p>
*
* @since SAX 1.0
* @author David Megginson,
*    <a href="mailto:sax@megginson.com">sax@megginson.com</a>
* @version 2.0
* @see org.xml.sax.Parser.setErrorHandler
* @see org.xml.sax.SAXParseException
*/

No license file was found, but licenses were detected in source scan.

// NO WARRANTY! This class is in the public domain.
/**
 * Interface for a list of XML attributes.
 *
 * <blockquote>
 * <em>This module, both source code and documentation, is in the
 * Public Domain, and comes with <strong>NO WARRANTY</strong>.</em>
 * </blockquote>
 *
 * This interface allows access to a list of attributes in
 * three different ways:
 *
 * <ol>
 * <li>by attribute index;</li>
 * <li>by Namespace-qualified name; or</li>
 * <li>by qualified (prefixed) name.</li>
 * </ol>
 *
 * The list will not contain attributes that were declared
 * #IMPLIED but not specified in the start tag. It will also not
 * contain attributes used as Namespace declarations (xmlns*) unless
 * the <code>http://xml.org/sax/features/namespace-prefixes</code>
 * feature is set to <var>true</var> (it is <var>false</var> by
 * default). </p>
 *
 * If the namespace-prefixes feature (see above) is <var>false</var>,
 * access by qualified name may not be available; if the
 * <code>http://xml.org/sax/features/namespaces</code>
 * feature is <var>false</var>, access by Namespace-qualified names
 * may not be available.</p>
This interface replaces the now-deprecated SAX1 \{ @link org.xml.sax.AttributeList AttributeList \} interface, which does not contain Namespace support. In addition to Namespace support, it adds the \texttt{getIndex()} methods (below).

The order of attributes in the list is unspecified, and will vary from implementation to implementation.

@since SAX 2.0
@author David Megginson, <a href="mailto:sax@megginson.com">sax@megginson.com</a>
@version 2.0
@see org.xml.sax.helpers.AttributeListImpl

Default implementation for AttributeList.

This module, both source code and documentation, is in the Public Domain, and comes with \textbf{NO WARRANTY}.

AttributeList implements the deprecated SAX1 \{ @link org.xml.saxAttributeList AttributeList \} interface, and has been replaced by the new SAX2 \{ @link org.xml.sax.helpers.AttributesImpl AttributesImpl \} interface.

This class provides a convenience implementation of the SAX \{ @link org.xml.sax.AttributeList AttributeList \} interface. This implementation is useful both for SAX parser writers, who can use it to provide attributes to the application, and for SAX application writers, who can use it to create a persistent copy of an element's attribute specifications:

```java
private AttributeList myatts;

public void startElement (String name, AttributeList atts)
{
    // create a persistent copy of the attribute list
    // for use outside this method
    myatts = new AttributeListImpl(atts);
```
Please note that SAX parsers are not required to use this class to provide an implementation of AttributeList; it is supplied only as an optional convenience. In particular, parser writers are encouraged to invent more efficient implementations.

@deprecated This class implements a deprecated interface, `AttributeList`; that interface has been replaced by `Attributes`, which is implemented in the `AttributesImpl` helper class.

@author David Megginson, sax@megginson.com
@version 2.0

An XML filter is like an XML reader, except that it obtains its events from another XML reader rather than a primary source like an XML document or database. Filters can modify a stream of events as they pass on to the final application.

The `XMLFilterImpl` helper class provides a convenient base for creating SAX2 filters, by passing on all `EntityResolver`, `DTDHandler`, `ContentHandler` and `ErrorHandler` events for creating SAX2 filters, by passing on all `EntityResolver`, `DTDHandler`, `ContentHandler` and `ErrorHandler`
* ErrorHandler} events automatically.</p>
*

* @since SAX 2.0
* @author David Megginson,
* <a href="mailto:sax@megginson.com">sax@megginson.com</a>
* @version 2.0
* @see org.xml.sax.helpers.XMLFilterImpl
*/

Found in path(s):
* /opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/org/xml/sax/XMLFilter.java

No license file was found, but licenses were detected in source scan.

/**
 * Interface for associating a SAX event with a document location.
 *
 * <blockquote>
 * <em>This module, both source code and documentation, is in the
 * Public Domain, and comes with <strong>NO WARRANTY</strong>.</em>
 * </blockquote>
 *
 * If a SAX parser provides location information to the SAX
 * application, it does so by implementing this interface and then
 * passing an instance to the application using the content
 * handler's [link org.xml.sax.ContentHandler#setDocumentLocator
 * setDocumentLocator] method. The application can use the
 * object to obtain the location of any other content handler event
 * in the XML source document.</p>
 *
 * Note that the results returned by the object will be valid only
 * during the scope of each content handler method: the application
 * will receive unpredictable results if it attempts to use the
 * locator at any other time.</p>
 *
 * SAX parsers are not required to supply a locator, but they are
 * very strongly encouraged to do so. If the parser supplies a
 * locator, it must do so before reporting any other document events.
 * If no locator has been set by the time the application receives
 * the [link org.xml.sax.ContentHandler#startDocument startDocument]
 * event, the application should assume that a locator is not
 * available.</p>
 *
 * @since SAX 1.0
 * @author David Megginson,
 * <a href="mailto:sax@megginson.com">sax@megginson.com</a>
 * @version 2.0
 * @see org.xml.sax.ContentHandler#setDocumentLocator
*/
/*
* Receive notification of basic DTD-related events.
*
* <blockquote>
* This module, both source code and documentation, is in the Public Domain, and comes with NO WARRANTY.
* </blockquote>
*
* If a SAX application needs information about notations and unparsed entities, then the application implements this interface and registers an instance with the SAX parser using the parser's setDTDHandler method. The parser uses the instance to report notation and unparsed entity declarations to the application.
*
* Note that this interface includes only those DTD events that the XML recommendation requires processors to report: notation and unparsed entity declarations.
*
* The SAX parser may report these events in any order, regardless of the order in which the notations and unparsed entities were declared; however, all DTD events must be reported after the document handler's startDocument event, and before the first startElement event.
*
* It is up to the application to store the information for future use (perhaps in a hash table or object tree). If the application encounters attributes of type "NOTATION", "ENTITY", or "ENTITIES", it can use the information that it obtained through this interface to find the entity and/or notation corresponding with the attribute value.
*
* @since SAX 1.0
* @author David Megginson,
* <a href="mailto:sax@megginson.com">sax@megginson.com</a>
* @version 2.0
* @see org.xml.sax.Parser#setDTDHandler
* @see org.xml.sax.HandlerBase
*/
No license file was found, but licenses were detected in source scan.

/*
 * The Apache Software License, Version 1.1
 * 
 * Copyright (c) 2001, 2002 The Apache Software Foundation.
 * All rights reserved.
 * 
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 
 * 1. Redistributions of source code must retain the above copyright
 * notice, this list of conditions and the following disclaimer.
 * 
 * 2. Redistributions in binary form must reproduce the above copyright
 * notice, this list of conditions and the following disclaimer in
 * the documentation and/or other materials provided with the
 * distribution.
 * 
 * 3. The end-user documentation included with the redistribution,
 * if any, must include the following acknowledgment:
 *    "This product includes software developed by the
 *    Apache Software Foundation (http://www.apache.org/)."
 *    Alternately, this acknowledgment may appear in the software itself,
 *    if and wherever such third-party acknowledgments normally appear.
 * 
 * 4. The names "Xerces" and "Apache Software Foundation" must
 *    not be used to endorse or promote products derived from this
 *    software without prior written permission. For written
 *    permission, please contact apache@apache.org.
 * 
 * 5. Products derived from this software may not be called "Apache",
 *    nor may "Apache" appear in their name, without prior written
 *    permission of the Apache Software Foundation.
 * 
 * THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
 * WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
 * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
 * DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
 * ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
 * SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
 * LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
 * USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
 * ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
 * OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
 * OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
* ====================================================================
* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation and was
* originally based on software copyright (c) 1999-2001, Sun Microsystems,
* Foundation, please see <http://www.apache.org/>.
* /

Found in path(s):
* /opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-
  jar/javax/xml/parsers/ConvertToURI.java
No license file was found, but licenses were detected in source scan.

/**
 * Default base class for handlers.
 *
 * <blockquote>
 * <em>This module, both source code and documentation, is in the
 * Public Domain, and comes with <strong>NO WARRANTY</strong>.</em>
 * </blockquote>
 *
 * This class implements the default behaviour for four SAX1
 * interfaces: EntityResolver, DTDHandler, DocumentHandler,
 * and ErrorHandler. It is now obsolete, but is included in SAX2 to
 * support legacy SAX1 applications. SAX2 applications should use
 * the [@link org.xml.sax.helpers.DefaultHandler DefaultHandler]
 * class instead.</p>
 *
 * Application writers can extend this class when they need to
 * implement only part of an interface; parser writers can
 * instantiate this class to provide default handlers when the
 * application has not supplied its own.</p>
 *
 * Note that the use of this class is optional.</p>
 *
 * @deprecated This class works with the deprecated
 * [@link org.xml.sax.DocumentHandler DocumentHandler]
 * interface. It has been replaced by the SAX2
 * [@link org.xml.sax.helpers.DefaultHandler DefaultHandler]
 * class.
 * @since SAX 1.0
 * @author David Megginson,
 * <a href="mailto:sax@megginson.com">sax@megginson.com</a>
 * @version 2.0
 * @see org.xml.sax.EntityResolver
 * @see org.xml.sax.DTDHandler
* @see org.xml.sax.DocumentHandler
* @see org.xml.sax.ErrorHandler
*/

Found in path(s):
* /opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/org/xml/sax/HandlerBase.java
No license file was found, but licenses were detected in source scan.

/*/ 
* Copyright (c) 2002 World Wide Web Consortium, 
* (Massachusetts Institute of Technology, Institut National de 
* Recherche en Informatique et en Automatique, Keio University). All 
* Rights Reserved. This program is distributed under the W3C's Software 
* Intellectual Property License. This program is distributed in the 
* hope that it will be useful, but WITHOUT ANY WARRANTY; without even 
* the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR 
* PURPOSE. 
* See W3C License http://www.w3.org/Consortium/Legal/ for more details. 
*/

Found in path(s):
* /opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/w3c/dom/ls/DOMBuilderFilter.java
* /opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/w3c/dom/ls/DOMInputSource.java
* /opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/w3c/dom/ls/DocumentLS.java
* /opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/w3c/dom/xpath/XPathNSResolver.java
* /opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/w3c/dom/xpath/XPathResult.java
* /opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/w3c/dom/xpath/XPathExpression.java
* /opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/w3c/dom/xpath/XPathException.java
* /opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/w3c/dom/ls/DOMEntityResolver.java
* /opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/w3c/dom/ls/DOMEntityResolver.java
* /opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/w3c/dom/ls/LSLoadEvent.java
* /opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/w3c/dom/ls/LSProgressEvent.java
* /opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/w3c/dom/ls/ParseErrorEvent.java
* /opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/w3c/dom/ls/DOMWriterFilter.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2002 World Wide Web Consortium,
 * (Massachusetts Institute of Technology, Institut National de
 * Recherche en Informatique et en Automatique, Keio University). All
 * Rights Reserved. This program is distributed under the W3C's Software
 * Intellectual Property License. This program is distributed in the
 * hope that it will be useful, but WITHOUT ANY WARRANTY; without even
 * the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR
 * PURPOSE.
 * See W3C License http://www.w3.org/Consortium/Legal/ for more details.
 */
/**
 * <strong>DOM Level 3 WD Experimental:
 * The DOM Level 3 specification is at the stage
 * of Working Draft, which represents work in
 * progress and thus may be updated, replaced,
 * or obsoleted by other documents at any time.</strong> <p>
 * <code>XPathEvaluator</code>, which will provide evaluation of XPath 1.0
 * expressions with no specialized extension functions or variables. It is
 * expected that the <code>XPathEvaluator</code> interface will be
 * implemented on the same object which implements the <code>Document</code>
 * interface in an implementation which supports the XPath DOM module.
 * <code>XPathEvaluator</code> implementations may be available from other
 * sources that may provide support for special extension functions or
 * variables which are not defined in this specification. The methods of
 * XPathExpression should be named with more-XPath-specific names because
 * the interface will often be implemented by the same object which
 * implements document. No change. The point of interfaces is to localize the
 * implementing namespace. This would make the method names unnecessarily
 * long and complex even though there are no conflicts in the interface
 * itself. The new core method getInterface is designed for discovering
 * interfaces of additional modules that may not be directly implemented on
 * the objects to which they are attached. This could be used to implement
 * XPath on a separate object. The user only refers to the separate
 * interfaces and not the proprietary aggregate implementation. Should entity
 * refs be supported so that queries can be made on them? No change. We will
 * not do this now. They are not part of the XPath data model. Note that
 * they may be present in the hierarchy of returned nodes, but may not
 * directly be requested or returned in the node set. What does createResult
 * create when one wants to reuse the XPath? It is not useful. Removed method.
 * Should ordering be a separate flag, or a type of result that can be
 * requested. As a type of result, it can be better optimized in
 * implementations. It makes sense as a type of result. Changed. Removed
 * method. Implementing XPathEvaluator on Document can be a problem due to
 * conflicts in the names of the methods. The working group finds no better
 * solution. GetInterface in Level 3 permits the object to be implemented
 */
separately. We should be committed to this. We will leave this issue open to see if we get more feedback on it. How does this interface adapt to XPath 2.0 and other query languages. No change. This interface is not intended to adapt to XPath 2.0 or other languages. The models of these are likely to be incompatible enough to require new APIs. For alternate implementations that can use this API, it can be obtained from different sources. Support for custom variables and functions would be very useful. No change. It is possible for an implementation to supply alternative sources of an XPathEvaluator that can be customized with a custom variable and function context. We do not specify how this is accomplished. It is too complex to address in this version of the XPath DOM.

See also the [Document Object Model (DOM) Level 3 XPath Specification](http://www.w3.org/TR/2002/WD-DOM-Level-3-XPath-20020328).

*/

Found in path(s):

* /opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/org/w3c/dom/xpath/XPathEvaluator.java

No license file was found, but licenses were detected in source scan.

/**
 * Interface for an element's attribute specifications.
 *
 * <blockquote>
 * <em>This module, both source code and documentation, is in the Public Domain, and comes with <strong>NO WARRANTY</strong>.</em>
 * </blockquote>
 *
 * This is the original SAX1 interface for reporting an element's attributes. Unlike the new \{@link org.xml.sax.Attributes Attributes\} interface, it does not support Namespace-related information.
 *
 * When an attribute list is supplied as part of a event, the list will return valid results only during the scope of the event; once the event handler returns control to the parser, the attribute list is invalid. To save a persistent copy of the attribute list, use the SAX1 \{@link org.xml.sax.helpers.AttributeListImpl AttributeListImpl\} helper class.
 *
 * An attribute list includes only attributes that have been specified or defaulted: #IMPLIED attributes will not be included.
 *
 * There are two ways for the SAX application to obtain information from the AttributeList. First, it can iterate through the entire list:
public void startElement (String name, AttributeList atts) {
    for (int i = 0; i < atts.getLength(); i++) {
        String name = atts.getName(i);
        String type = atts.getType(i);
        String value = atts.getValue(i);
        [...]
    }
}
</pre>

(Note that the result of getLength() will be zero if there
are no attributes.)

As an alternative, the application can request the value or
type of specific attributes:

public void startElement (String name, AttributeList atts) {
    String identifier = atts.getValue("id");
    String label = atts.getValue("label");
    [...]
}
</pre>

@deprecated This interface has been replaced by the SAX2
    { @link org.xml.sax.Attributes Attributes }
    interface, which includes Namespace support.
@since SAX 1.0
@author David Megginson,
<a href="mailto:sax@megginson.com">sax@megginson.com</a>
@version 2.0
@see org.xml.sax.DocumentHandler#startElement startElement
@see org.xml.sax.helpers.AttributeListImpl AttributeListImpl
*/

Found in path(s):
/opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/org/xml/sax/AttributeList.java
No license file was found, but licenses were detected in source scan.

/**
 * Receive notification of general document events.
 *
 * This module, both source code and documentation, is in the
 * Public Domain, and comes with NO WARRANTY.</em>
 */
This was the main event-handling interface for SAX1; in
SAX2, it has been replaced by `ContentHandler`, which provides Namespace support and reporting
of skipped entities. This interface is included in SAX2 only
to support legacy SAX1 applications.

The order of events in this interface is very important, and
mirrors the order of information in the document itself. For
example, all of an element's content (character data, processing
instructions, and/or subelements) will appear, in order, between
the startElement event and the corresponding endElement event.

Application writers who do not want to implement the entire
interface can derive a class from `HandlerBase`, which implements
the default functionality; parser writers can instantiate
`HandlerBase` to obtain a default handler. The application can find
the location of any document event using the Locator interface
supplied by the Parser through the setDocumentLocator method.

@deprecated This interface has been replaced by the SAX2
`ContentHandler` interface, which includes Namespace support.
@since SAX 1.0
@author David Megginson,
<a href="mailto:sax@megginson.com">sax@megginson.com</a>
@version 2.0
@see org.xml.sax.Parser#setDocumentHandler
@see org.xml.sax.Locator
@see org.xml.sax.HandlerBase
*/

// NO WARRANTY! This class is in the Public Domain.
/**
 * Factory for creating an XML reader.
 *
 * This module, both source code and documentation, is in the
 * Public Domain, and comes with <strong>NO WARRANTY</strong>. See <a href="http://www.saxproject.org">http://www.saxproject.org</a>
 * for further information.
 */
This class contains static methods for creating an XML reader from an explicit class name, or based on runtime defaults:

```
try {
  XMLReader myReader = XMLReaderFactory.createXMLReader();
} catch (SAXException e) {
  System.err.println(e.getMessage());
}
```

Note to Distributions bundled with parsers:
You should modify the implementation of the no-arguments createXMLReader to handle cases where the external configuration mechanisms aren't set up. That method should do its best to return a parser when one is in the class path, even when nothing bound its class name to org.xml.sax.driver so those configuration mechanisms would see it.

@since SAX 2.0
@author David Megginson, David Brownell
@version 2.0r2pre3

This class is in the Public Domain. NO WARRANTY!

Encapsulate Namespace logic for use by SAX drivers.

This module, both source code and documentation, is in the Public Domain, and comes with NO WARRANTY.

This class encapsulates the logic of Namespace processing:
  it tracks the declarations currently in force for each context
  and automatically processes qualified XML 1.0 names into their Namespace parts; it can also be used in reverse for generating XML 1.0 from Namespaces.

Namespace support objects are reusable, but the reset method must be invoked between each session.

Here is a simple session:
* <pre>
* String parts[] = new String[3];
* NamespaceSupport support = new NamespaceSupport();
*
* support.pushContext();
* support.declarePrefix("", "http://www.w3.org/1999/xhtml");
* support.declarePrefix("dc", "http://www.purl.org/dc#");
*
* String parts[] = support.processName("p", parts, false);
* System.out.println("Namespace URI: " + parts[0]);
* System.out.println("Local name: " + parts[1]);
* System.out.println("Raw name: " + parts[2]);

* String parts[] = support.processName("dc:title", parts, false);
* System.out.println("Namespace URI: " + parts[0]);
* System.out.println("Local name: " + parts[1]);
* System.out.println("Raw name: " + parts[2]);

* support.popContext();
* </pre>

* <p>Note that this class is optimized for the use case where most
* elements do not contain Namespace declarations: if the same
* prefix/URI mapping is repeated for each context (for example), this
* class will be somewhat less efficient.</p>
*
* @since SAX 2.0
* @author David Megginson,
*         <a href="mailto:sax@megginson.com">sax@megginson.com</a>
* @version 2.0
*/

Found in path(s):
* /opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-
jar/org/xml/sax/helpers/NamespaceSupport.java
No license file was found, but licenses were detected in source scan.

// NO WARRANTY! This class is in the public domain.
/**
* Default base class for SAX2 event handlers.
* *
* <blockquote>
* <em>This module, both source code and documentation, is in the
* Public Domain, and comes with <strong>NO WARRANTY</strong>.</em>
* </blockquote>
* 
* <p>This class is available as a convenience base class for SAX2
* applications: it provides default implementations for all of the
* callbacks in the four core SAX2 handler classes:</p>
*
<br><ul>
<li>{@link org.xml.sax.EntityResolver EntityResolver}</li>
<li>{@link org.xml.sax.DTDHandler DTDHandler}</li>
<li>{@link org.xml.sax.ContentHandler ContentHandler}</li>
<li>{@link org.xml.sax.ErrorHandler ErrorHandler}</li>
</ul>
*
*p> Application writers can extend this class when they need to
* implement only part of an interface; parser writers can
* instantiate this class to provide default handlers when the
* application has not supplied its own.</p>
*
*p> This class replaces the deprecated SAX1
* {@link org.xml.sax.HandlerBase HandlerBase} class.</p>
*
@since SAX 2.0
* @author David Megginson,
*        <a href="mailto:sax@megginson.com">sax@megginson.com</a>
* @version 2.0
* @see org.xml.sax.EntityResolver
* @see org.xml.sax.DTDHandler
* @see org.xml.sax.ContentHandler
* @see org.xml.sax.ErrorHandler
*/

Found in path(s):
* /opt/colas/permits/1150916854_1617719483.11/0/xml-apis-2.0-2-sources-2-
jar/org/xml/sax/helpers/DefaultHandler.java
No license file was found, but licenses were detected in source scan.

/**
* Encapsulate an XML parse error or warning.
*
* <em>This module, both source code and documentation, is in the
* Public Domain, and comes with <strong>NO WARRANTY</strong>. </em>
* 
*/

*p> This exception will include information for locating the error
* in the original XML document. Note that although the application
* will receive a SAXParseException as the argument to the handlers
* in the [ @link org.xml.sax.ErrorHandler ErrorHandler ] interface,
* the application is not actually required to throw the exception;
* instead, it can simply read the information in it and take a
* different action.</p>
* "Since this exception is a subclass of [ @link org.xml.sax.SAXException
*SAXException], it inherits the ability to wrap another exception."</p>
*
* @since SAX 1.0
* @author David Megginson,
* <a href="mailto:sax@megginson.com">sax@megginson.com</a>
* @version 2.0
* @see org.xml.sax.SAXException
* @see org.xml.sax.Locator
* @see org.xml.sax.ErrorHandler
*/

Found in path(s):
*/opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-
jar/org/xml/sax/SAXParseException.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2000 World Wide Web Consortium,
* (Massachusetts Institute of Technology, Institut National de
* Recherche en Informatique et en Automatique, Keio University). All
* Rights Reserved. This program is distributed under the W3C's Software
* Intellectual Property License. This program is distributed in the
* hope that it will be useful, but WITHOUT ANY WARRANTY; without even
* the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR
* PURPOSE.
* See W3C License http://www.w3.org/Consortium/Legal/ for more details.
*/

Found in path(s):
*/opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-
jar/org/w3c/dom/ranges/RangeException.java
*/opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/org/w3c/dom/CharacterData.java
*/opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/org/w3c/dom/NamedNodeMap.java
*/opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/org/w3c/dom/Text.java
*/opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/org/w3c/dom/events/Event.java
*/opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/org/w3c/dom/events/DocumentEvent.java
*/opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/org/w3c/dom/events/EventTarget.java
*/opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/org/w3c/dom/ProcessingInstruction.java
* /opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/org/w3c/dom/events/EventException.java
* /opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/org/w3c/dom/traversal/TreeWalker.java
* /opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/org/w3c/dom/events/MutationEvent.java
* /opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/org/w3c/dom/Comment.java
* /opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/org/w3c/dom/EntityReference.java
* /opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/org/w3c/dom/DocumentFragment.java
* /opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/org/w3c/dom/CDATASection.java
* /opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/org/w3c/dom/Node.java
* /opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/org/w3c/dom/NodeList.java
* /opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/org/w3c/dom/DocumentType.java
* /opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/org/w3c/dom/DOMImplementation.java
* /opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/org/w3c/dom/events/EventListener.java
* /opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/org/w3c/dom/traversal/NodeIterator.java
* /opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/org/w3c/dom/Entity.java
* /opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/org/w3c/dom/DOMException.java
* /opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/org/w3c/dom/Element.java
* /opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/org/w3c/dom/ranges/Range.java
* /opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/org/w3c/dom/traversal/NodeFilter.java

No license file was found, but licenses were detected in source scan.

/**
 * Basic interface for resolving entities.
 *<blockquote>
 *<em>This module, both source code and documentation, is in the
 * Public Domain, and comes with <strong>NO WARRANTY</strong>.</em>
 *</blockquote>
 *<p>If a SAX application needs to implement customized handling
 *for external entities, it must implement this interface and
 *register an instance with the SAX driver using the
 *{@link org.xml.sax.XMLReader#setEntityResolver setEntityResolver}
 *method.</p>
 *<p>The XML reader will then allow the application to intercept any
 *external entities (including the external DTD subset and external
 *parameter entities, if any) before including them.</p>
Many SAX applications will not need to implement this interface, but it will be especially useful for applications that build XML documents from databases or other specialised input sources, or for applications that use URI types other than URLs.

The following resolver would provide the application with a special character stream for the entity with the system identifier "http://www.myhost.com/today":

```java
import org.xml.sax.EntityResolver;
import org.xml.sax.InputSource;

public class MyResolver implements EntityResolver {
  public InputSource resolveEntity (String publicId, String systemId)
  {
    if (systemId.equals("http://www.myhost.com/today")) {
      // return a special input source
      MyReader reader = new MyReader();
      return new InputSource(reader);
    } else {
      // use the default behaviour
      return null;
    }
  }
}
```

The application can also use this interface to redirect system identifiers to local URIs or to look up replacements in a catalog (possibly by using the public identifier).

@since SAX 1.0
@author David Megginson,
<a href="mailto:sax@megginson.com">sax@megginson.com</a>
@version 2.0
@see org.xml.sax.Parser#setEntityResolver
@see org.xml.sax.InputSource

Found in path(s):
/opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/org/xml/sax/EntityResolver.java
No license file was found, but licenses were detected in source scan.
documentation contained in this distribution into the Public Domain.

Found in path(s):
/*
 * Copyright (c) 2000 World Wide Web Consortium,
 * (Massachusetts Institute of Technology, Institut National de
 * Recherche en Informatique et en Automatique, Keio University). All
 * Rights Reserved. This program is distributed under the W3C's Software
 * Intellectual Property License. This program is distributed in the
 * hope that it will be useful, but WITHOUT ANY WARRANTY; without even
 * the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR
 * PURPOSE. See W3C License http://www.w3.org/Consortium/Legal/ for more
 * details.
 */

Found in path(s):
*/opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/org/w3c/dom/html/HTMLElement.w3c
*/opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/org/w3c/dom/html/HTMLDocument.w3c
*/opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/org/w3c/dom/html/HTMLBodyElement.w3c
*/opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/org/w3c/dom/html/HTMLBRElement.w3c
*/opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/org/w3c/dom/html/HTMLLabelElement.w3c
*/opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/org/w3c/dom/html/HTMLFrameElement.w3c
*/opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/org/w3c/dom/html/HTMLTableCellElement.w3c
*/opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/org/w3c/dom/html/HTMLDocument.w3c
*/opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/org/w3c/dom/html/HTMLTableRowElement.w3c
*/opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/org/w3c/dom/html/HTMLSelectElement.w3c
*/opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/org/w3c/dom/html/HTMLMenuElement.w3c
*/opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/org/w3c/dom/html/HTMLInputElement.w3c
*/opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/org/w3c/dom/html/HTMLHeadingElement.w3c
*/opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/org/w3c/dom/html/HTMLDirectoryElement.w3c
*/opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/org/w3c/dom/html/HTMLTableCellElement.w3c
*/opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-
No license file was found, but licenses were detected in source scan.

/**
 * Provide an optional convenience implementation of Locator.
 *<em>This module, both source code and documentation, is in the Public Domain, and comes with NO WARRANTY.</em>
 */

<p>This class is available mainly for application writers, who can use it to make a persistent snapshot of a locator at any point during a document parse:</p>
* Locator locator;
* Locator startloc;
*
* public void setLocator (Locator locator)
* {
*   // note the locator
*   this.locator = locator;
* }
*
* public void startDocument ()
* {
*   // save the location of the start of the document
*   // for future use.
*   Locator startloc = new LocatorImpl(locator);
* }
* </pre>
*
* <p>Normally, parser writers will not use this class, since it
* is more efficient to provide location information only when
* requested, rather than constantly updating a Locator object.</p>
*
* @since SAX 1.0
* @author David Megginson,
*       <a href="mailto:sax@megginson.com">sax@megginson.com</a>
* @version 2.0
* @see org.xml.sax.Locator Locator
* /

Found in path(s):
* /opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-
jar/org/xml/sax/helpers/LocatorImpl.java
No license file was found, but licenses were detected in source scan.

/*
* The Apache Software License, Version 1.1
*
* Copyright (c) 2001 The Apache Software Foundation. All rights
* reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
*
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
*
* 2. Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in
the documentation and/or other materials provided with the
distribution.

3. The end-user documentation included with the redistribution,
if any, must include the following acknowledgment:
"This product includes software developed by the
Apache Software Foundation (http://www.apache.org/)."
Alternately, this acknowledgment may appear in the software itself,
if and wherever such third-party acknowledgments normally appear.

4. The name "Apache Software Foundation" must not be used to endorse or
promote products derived from this software without prior written
permission. For written permission, please contact apache@apache.org.

5. Products derived from this software may not be called "Apache",
nor may "Apache" appear in their name, without prior written
permission of the Apache Software Foundation.

THIS SOFTWARE IS PROVIDED `"AS IS" AND ANY EXPRESSED OR IMPLIED
WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
 SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
 OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.

This software consists of voluntary contributions made by many
individuals on behalf of the Apache Software Foundation and was
originally based on software copyright (c) 1999-2001, Sun Microsystems,
Foundation, please see <http://www.apache.org/>.

Found in path(s):
* /opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-
jar/javax/xml/parsers/ParserConfigurationException.java
* /opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-
jar/javax/xml/parsers/FactoryFinder.java
* /opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-
jar/javax/xml/parsers/FactoryConfigurationError.java
* /opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-
// NO WARRANTY!  This class is in the Public Domain.
/**
 * Interface for reading an XML document using callbacks.
 *
 * <blockquote>
 * This module, both source code and documentation, is in the
 * Public Domain, and comes with <strong>NO WARRANTY</strong>.<br />
 * </blockquote>
 *
 * <p>Note: despite its name, this interface does</p>
 * extend the standard Java {@link java.io.Reader Reader} interface, because reading XML is a fundamentally different activity
 * than reading character data.</p>
 *
 * XMLReader is the interface that an XML parser's SAX2 driver must
 * implement.  This interface allows an application to set and
 * query features and properties in the parser, to register
 * event handlers for document processing, and to initiate
 * a document parse.</p>
 *
 * All SAX interfaces are assumed to be synchronous: the
 * [parse] methods must not return until parsing
 * is complete, and readers must wait for an event-handler callback
 * to return before reporting the next event.</p>
 *
 * This interface replaces the (now deprecated) SAX 1.0 [Parser] interface. The XMLReader interface
 * contains two important enhancements over the old Parser
 * interface:</p>
 *
 * it adds a standard way to query and set features and
 * properties; and</li>
 * it adds Namespace support, which is required for many
 * higher-level XML standards.</li>
 * </ol>
 *
 * There are adapters available to convert a SAX1 Parser to
 * a SAX2 XMLReader and vice-versa.</p>
A single input source for an XML entity.

This class allows a SAX application to encapsulate information about an input source in a single object, which may include:
- a public identifier,
- a system identifier,
- a byte stream (possibly with a specified encoding),
- and/or a character stream.

There are two places that the application will deliver this input source to the parser: as the argument to the Parser.parse method, or as the return value of the EntityResolver.resolveEntity method.

The SAX parser will use the InputSource object to determine how to read XML input. If there is a character stream available, the parser will read that stream directly; if not, the parser will use a byte stream, if available; if neither a character stream nor a byte stream is available, the parser will attempt to open a URI connection to the resource identified by the system identifier.

An InputSource object belongs to the application: the SAX parser shall never modify it in any way (it may modify a copy if necessary).
Found in path(s):
* /opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/org/xml/sax/InputSource.java
No license file was found, but licenses were detected in source scan.

// NO WARRANTY! This class is in the Public Domain.
/**
 * Exception class for an unsupported operation.
 *
 * <p>An XMLReader will throw this exception when it recognizes a
 * feature or property identifier, but cannot perform the requested
 * operation (setting a state or value). Other SAX2 applications and
 * extensions may use this class for similar purposes.</p>
 *
 * @since SAX 2.0
 * @author David Megginson,
 *         <a href="mailto:sax@megginson.com">sax@megginson.com</a>
 * @version 2.0
 * @see org.xml.sax.SAXNotSupportedException
 */

Found in path(s):
* /opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/org/xml/sax/SAXNotSupportedException.java
No license file was found, but licenses were detected in source scan.

/*
 * The Apache Software License, Version 1.1
 *
 * Copyright (c) 2002 The Apache Software Foundation. All rights
 * reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 *
 * 1. Redistributions of source code must retain the above copyright
 *    notice, this list of conditions and the following disclaimer.

 /*
* 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* 3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment: "This product includes software developed by the Apache Software Foundation (http://www.apache.org/)."
* Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear.

* 4. The name "Apache Software Foundation" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact apache@apache.org.

* 5. Products derived from this software may not be called "Apache", nor may "Apache" appear in their name, without prior written permission of the Apache Software Foundation.

* THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

* =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=

* This software consists of voluntary contributions made by many individuals on behalf of the Apache Software Foundation and was originally based on software copyright (c) 1999-2002, Sun Microsystems, Inc., http://www.sun.com. For more information on the Apache Software Foundation, please see <http://www.apache.org/>.

* /

Found in path(s):
* /opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-
// NO WARRANTY! This class is in the public domain.
/**
* Adapt a SAX1 Parser as a SAX2 XMLReader.
*
* @since SAX 2.0
* @author David Megginson,
*         <a href="mailto:sax@megginson.com">sax@megginson.com</a>
* @version 2.0
* @see org.xml.sax.helpers.XMLReaderAdapter
* @see org.xml.sax.XMLReader
*/
/**
* Adapt a SAX1 AttributeList as a SAX2 Attributes object.
*
This class is in the Public Domain, and comes with NO WARRANTY of any kind.</p>

This wrapper class is used only when Namespace support is disabled -- it provides pretty much a direct mapping from SAX1 to SAX2, except that names and types are interned whenever requested.</p>

Default implementation of the Attributes interface.

This module, both source code and documentation, is in the Public Domain, and comes with NO WARRANTY.</p>


This class provides a default implementation of the SAX2 Attributes interface, with the addition of manipulators so that the list can be modified or reused.

There are two typical uses of this class:

- to take a persistent snapshot of an Attributes object in a startElement event; or
- to construct or modify an Attributes object in a SAX2 driver or filter.

This class replaces the now-deprecated SAX1 AttributeListImpl class; in addition to supporting the updated Attributes interface rather than the deprecated AttributeList interface, it also includes a much more efficient implementation using a single array rather than a set of Vectors.

@since SAX 2.0

@since SAX 2.0

@version 2.0.1 (sax2r2)
/**
 * Create a new instance of a class by name.
 *
 * <blockquote>
 * <em>This module, both source code and documentation, is in the
 * Public Domain, and comes with <strong>NO WARRANTY</strong>. </em>
 * See <a href='http://www.saxproject.org'>http://www.saxproject.org</a>
 * for further information.
 * 
 * </blockquote>

 * This class contains a static method for creating an instance of a
 * class from an explicit class name. It tries to use the thread's context
 * ClassLoader if possible and falls back to using
 * Class.forName(String). It also takes into account JDK 1.2+'s
 * AccessController mechanism for performing its actions. </p>

 * This code is designed to run on JDK version 1.1 and later and compile
 * on versions of Java 2 and later. </p>

 * @author Edwin Goei, David Brownell, Neil Graham
 * @version $Id: NewInstance.java,v 1.2 2002/08/26 23:55:45 neilg Exp $
 */

/**
 * Encapsulate a general SAX error or warning.
 *
 * <blockquote>
 * This module, both source code and documentation, is in the
 * Public Domain, and comes with <strong>NO WARRANTY</strong>. </em>
 * 
 * </blockquote>

 * This class can contain basic error or warning information from
 * either the XML parser or the application: a parser writer or
 * application writer can subclass it to provide additional
 * functionality. SAX handlers may throw this exception or
* any exception subclassed from it.</p>
*  
* <p>If the application needs to pass through other types of
* exceptions, it must wrap those exceptions in a SAXException
* or an exception derived from a SAXException.</p>
*  
* <p>If the parser or application needs to include information about a
* specific location in an XML document, it should use the
* [@link org.xml.sax.SAXParseException SAXParseException] subclass.</p>
*  
* @since SAX 1.0
* @author David Megginson,
* @version 2.0
* @see org.xml.sax.SAXParseException
*/

Found in path(s):
* /opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/org/xml/sax/SAXException.java

No license file was found, but licenses were detected in source scan.

// NO WARRANTY! This class is in the Public Domain.
/**
 * Exception class for an unrecognized identifier.
 *
 * &lt;blockquote&gt;
 * &lt;em&gt;This module, both source code and documentation, is in the
 * Public Domain, and comes with &lt;strong&gt;NO WARRANTY&lt;/strong&gt;.&lt;/em&gt;
 * &lt;/blockquote&gt;
 *
* <p>An XMLReader will throw this exception when it finds an
* unrecognized feature or property identifier; SAX applications and
* extensions may use this class for other, similar purposes.</p>
*
* @since SAX 2.0
* @author David Megginson,
* @version 2.0
* @see org.xml.sax.SAXNotSupportedException
*/

Found in path(s):
* /opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/org/xml/sax/SAXNotRecognizedException.java

No license file was found, but licenses were detected in source scan.

because SAX is released to the public domain, there is no warranty
merchantability and fitness for a particular purpose. The entire
into the public domain.
Public Domain. SAX comes with NO WARRANTY or guarantee of fitness

Found in path(s):
* /opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/org/xml/sax/package.html
No license file was found, but licenses were detected in source scan.

/ *
* Copyright (c) 2002 World Wide Web Consortium,
* (Massachusetts Institute of Technology, Institut National de
* Recherche en Informatique et en Automatique, Keio University). All
* Rights Reserved. This program is distributed under the W3C's Software
* Intellectual Property License. This program is distributed in the
* hope that it will be useful, but WITHOUT ANY WARRANTY; without even
* the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR
* PURPOSE.
* See W3C License http://www.w3.org/Consortium/Legal/ for more details.
*/
/**
 * Create a new <code>DOMBuilder</code>. The newly constructed parser may
 * then be configured by means of its <code>setFeature</code> method,
 * and used to parse documents by means of its <code>parse</code>
 * method.
 * @param mode  The <code>mode</code> argument is either
 *   <code>MODE_SYNCHRONOUS</code> or <code>MODE_ASYNCHRONOUS</code>, if
 *   <code>mode</code> is <code>MODE_SYNCHRONOUS</code> then the
 *   <code>DOMBuilder</code> that is created will operate in synchronous
 *   mode, if it's <code>MODE_ASYNCHRONOUS</code> then the
 *   <code>DOMBuilder</code> that is created will operate in
 *   asynchronous mode.
 * @param schemaType  An absolute URI representing the type of the schema
 *   language used during the load of a <code>Document</code> using the
 *   newly created <code>DOMBuilder</code>. Note that no lexical
 *   checking is done on the absolute URI. In order to create a
 *   <code>DOMBuilder</code> for any kind of schema types (i.e. the
 *   DOMBuilder will be free to use any schema found), use the value
 *   <code>null</code>. For W3C XML Schema, applications must use the
 *   value <code>"http://www.w3.org/2001/XMLSchema"</code>. For XML DTD
 *   , applications must use the value
 *   <code>"http://www.w3.org/TR/REC-xml"</code>. Other Schema languages
 *   are outside the scope of the W3C and therefore should recommend an
 *   absolute URI in order to use this method.
 * @return  The newly created <code>DOMBuilder</code> object. This
 *   <code>DOMBuilder</code> is either synchronous or asynchronous
 *   depending on the value of the <code>mode</code> argument. By
 *   default, the newly created <code>DOMBuilder</code> does not contain
 *   a <code>DOMErrorHandler</code>, i.e. the value of the
* <code>errorHandler</code> is <code>null</code>. However, implementations may provide a default error handler at creation time. In that case, the initial value of the <code>errorHandler</code> attribute on the new created <code>DOMBuilder</code> contains a reference to the default error handler.

* @exception DOMException

* NOT_SUPPORTED_ERR: Raised if the requested mode or schema type is not supported.

*/

Found in path(s):
* /opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/org/w3c/dom/ls/DOMImplementationLS.java
No license file was found, but licenses were detected in source scan.

/**
* Basic interface for SAX (Simple API for XML) parsers.
*
* <blockquote>
* <em>This module, both source code and documentation, is in the Public Domain, and comes with <strong>NO WARRANTY</strong>.</em>
* </blockquote>
*
* This was the main event supplier interface for SAX1; it has been replaced in SAX2 by [@link org.xml.sax.XMLReader XMLReader], which includes Namespace support and sophisticated configurability and extensibility.</p>
*
* All SAX1 parsers must implement this basic interface: it allows applications to register handlers for different types of events and to initiate a parse from a URI, or a character stream.</p>
*
* All SAX1 parsers must also implement a zero-argument constructor (though other constructors are also allowed).</p>
*
* SAX1 parsers are reusable but not re-entrant: the application may reuse a parser object (possibly with a different input source) once the first parse has completed successfully, but it may not invoke the parse() methods recursively within a parse.</p>
*
* @deprecated This interface has been replaced by the SAX2 interface, which includes Namespace support.
* @since SAX 1.0
* @author David Megginson,
*         <a href="mailto:sax@megginson.com">sax@megginson.com</a>
* @version 2.0
/* @see org.xml.sax.EntityResolver
 * @see org.xml.sax.DTDHandler
 * @see org.xml.sax.DocumentHandler
 * @see org.xml.sax.ErrorHandler
 * @see org.xml.sax.HandlerBase
 * @see org.xml.sax.InputSource
 */

Found in path(s):
/ /opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/org/xml/sax/Parser.java
No license file was found, but licenses were detected in source scan.

// Public Domain: no warranty.
/**
 * SAX2 extension handler for DTD declaration events.
 *
 * <blockquote>
 * <em>This module, both source code and documentation, is in the
 * Public Domain, and comes with <strong>NO WARRANTY </strong>.</em>
 * </blockquote>
 *
 * This is an optional extension handler for SAX2 to provide
 * information about DTD declarations in an XML document. XML
 * readers are not required to support this handler, and this
 * handler is not included in the core SAX2 distribution.</p>
 *
 * Note that data-related DTD declarations (unparsed entities and
 * notations) are already reported through the { @link
 * org.xml.sax.DTDHandler DTDHandler} interface.</p>
 *
 * If you are using the declaration handler together with a lexical
 * handler, all of the events will occur between the
 * { @link org.xml.sax.ext.LexicalHandler#startDTD startDTD} and the
 * { @link org.xml.sax.ext.LexicalHandler#endDTD endDTD} events.</p>
 *
 * To set the DeclHandler for an XML reader, use the
 * { @link org.xml.sax.XMLReader#setProperty setProperty} method
 * with the propertyId "http://xml.org/sax/properties/declaration-handler".
 * If the reader does not support declaration events, it will throw a
 * { @link org.xml.sax.SAXNotRecognizedException SAXNotRecognizedException}
 * or a
 * { @link org.xml.sax.SAXNotSupportedException SAXNotSupportedException}
 * when you attempt to register the handler.</p>
 *
 * @since 1.0
 * @author David Megginson,
 * <a href="mailto:sax@megginson.com">sax@megginson.com</a>
 * @version 1.0
* @see org.xml.sax.XMLReader
*

Found in path(s):
* /opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/org/xml/sax/ext/DeclHandler.java
No license file was found, but licenses were detected in source scan.

/**
 * Java-specific class for dynamically loading SAX parsers.
 * <p>
 * <blockquote>
 * <em>This module, both source code and documentation, is in the Public Domain, and comes with <strong>NO WARRANTY</strong>.</em>
 * </blockquote>
 * <p>Note: This class is designed to work with the now-deprecated SAX1 <a href='http://www.saxproject.org'>http://www.saxproject.org</a> class. SAX2 applications should use <a href='http://www.saxproject.org'>http://www.saxproject.org</a> instead.
 * <p>ParserFactory is not part of the platform-independent definition of SAX; it is an additional convenience class designed specifically for Java XML application writers. SAX applications can use the static methods in this class to allocate a SAX parser dynamically at run-time based either on the value of the `org.xml.sax.parser' system property or on a string containing the class name.
 * <p>Note that the application still requires an XML parser that implements SAX1.
 * <p>@deprecated This class works with the deprecated { @link org.xml.sax.Parser Parser} interface.
 * @since SAX 1.0
 * @author David Megginson
 * @version 2.0r2pre3
 */

Found in path(s):
* /opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/org/xml/sax/helpers/ParserFactory.java
No license file was found, but licenses were detected in source scan.

/*
 * The Apache Software License, Version 1.1
 */
Copyright (c) 2002 The Apache Software Foundation.
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

1. Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in
the documentation and/or other materials provided with the
distribution.

3. The end-user documentation included with the redistribution,
if any, must include the following acknowledgment:
"This product includes software developed by the
Apache Software Foundation (http://www.apache.org/)."
Alternately, this acknowledgment may appear in the software itself,
if and wherever such third-party acknowledgments normally appear.

4. The names "Xerces" and "Apache Software Foundation" must
not be used to endorse or promote products derived from this
software without prior written permission. For written
permission, please contact apache@apache.org.

5. Products derived from this software may not be called "Apache",
nor may "Apache" appear in their name, without prior written
permission of the Apache Software Foundation.

THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.

This software consists of voluntary contributions made by many
individuals on behalf of the Apache Software Foundation and was
* originally based on software copyright (c) 1999, International
* Business Machines, Inc., http://www.apache.org. For more
* information on the Apache Software Foundation, please see
*/

Found in path(s):
* /opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-
jar/javax/xml/parsers/FilePathToURI.java
No license file was found, but licenses were detected in source scan.

// NO WARRANTY! This class is in the public domain.
/**
 * Adapt a SAX2 XMLReader as a SAX1 Parser.
 *
 * <blockquote>
 * <em>This module, both source code and documentation, is in the
 * Public Domain, and comes with <strong>NO WARRANTY</strong>.</em>
 * </blockquote>
 *
 * @since SAX 2.0
 * @author David Megginson,
 *         <a href="mailto:sax@megginson.com">sax@megginson.com</a>
 * @version 2.0
 * @see org.xml.sax.Parser
 * @see org.xml.sax.XMLReader
*/

Found in path(s):
* /opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-
jar/org/xml/sax/helpers/XMLReaderAdapter.java
No license file was found, but licenses were detected in source scan.

// NO WARRANTY! This class is in the public domain.
/**
 * Receive notification of the logical content of a document.
 *
 * <blockquote>
 * <em>This module, both source code and documentation, is in the
 * Public Domain, and comes with <strong>NO WARRANTY</strong>.</em>
 * </blockquote>
<p>This is the main interface that most SAX applications implement: if the application needs to be informed of basic parsing events, it implements this interface and registers an instance with the SAX parser using the <code>@link org.xml.sax.XMLReader#setContentHandler setContentHandler</code> method. The parser uses the instance to report basic document-related events like the start and end of elements and character data.</p>

<p>The order of events in this interface is very important, and mirrors the order of information in the document itself. For example, all of an element's content (character data, processing instructions, and/or subelements) will appear, in order, between the startElement event and the corresponding endElement event.</p>

<p>This interface is similar to the now-deprecated SAX 1.0 DocumentHandler interface, but it adds support for Namespaces and for reporting skipped entities (in non-validating XML processors).</p>

<p>Implementors should note that there is also a Java class <code>@link java.net.ContentHandler ContentHandler</code> in the java.net package; that means that it's probably a bad idea to do.</p>

<p>In fact, "import ...*" is usually a sign of sloppy programming anyway, so the user should consider this a feature rather than a bug.</p>

<p>@since SAX 2.0</p>

<p>@author David Megginson, <a href="mailto:sax@megginson.com">sax@megginson.com</a></p>

<p>@version 2.0</p>

<p>@see org.xml.sax.XMLReader</p>

<p>@see org.xml.sax.DTDHandler</p>

<p>@see org.xml.sax.ErrorHandler</p>

/*

Found in path(s):
* /opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/org/xml/sax/ContentHandler.java
No license file was found, but licenses were detected in source scan.

// Public Domain: no warranty.
/**
 * SAX2 extension handler for lexical events.
 *
 * <blockquote>
 * <em>This module, both source code and documentation, is in the
 * Public Domain, and comes with <strong>NO WARRANTY</strong>.
 * </em>
 * </blockquote>
 *
 * This is an optional extension handler for SAX2 to provide
 * lexical information about an XML document, such as comments
 * and CDATA section boundaries; XML readers are not required to
 * support this handler, and it is not part of the core SAX2
 * distribution.</p>
 *
 * The events in the lexical handler apply to the entire document,
 * not just to the document element, and all lexical handler events
 * must appear between the content handler's startDocument and
 * endDocument events.</p>
 *
 * To set the LexicalHandler for an XML reader, use the
 * {@link org.xml.sax.XMLReader#setProperty setProperty} method
 * with the propertyId "http://xml.org/sax/properties/lexical-handler".
 * If the reader does not support lexical events, it will throw a
 * SAXNotRecognizedException or a
 * SAXNotSupportedException when you attempt to register the handler.</p>
 *
 * @since 1.0
 * @author David Megginson,
 * <a href="mailto:sax@megginson.com">sax@megginson.com</a>
 * @version 1.0
 * @see org.xml.sax.XMLReader#setProperty
 * @see org.xml.sax.SAXNotRecognizedException
 * @see org.xml.sax.SAXNotSupportedException
 */

Found in path(s):
* /opt/cola/permits/1150916854_1617719483.11/0/xml-apis-2-0-2-sources-2-jar/org/xml/sax/ext/LexicalHandler.java

1.4 netty-codec-http 4.1.69.Final

1.4.1 Available under license:
   No license file was found, but licenses were detected in source scan.

/
* Copyright 2014 The Netty Project
* 
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* 
* https://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/websocketx/extensions/compression/PerMessageDeflateDecoder.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/websocketx/extensions/compression/PerMessageDeflateEncoder.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/websocketx/extensions/WebSocketExtensionUtil.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/spdy/SpdyHeaderBlockRawDecoder.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/HttpChunkedInput.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/websocketx/extensions/WebSocketExtensionDecoder.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/spdy/SpdyFrameEncoder.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/websocketx/extensions/WebSocketServerExtension.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/websocketx/extensions/WebSocketServerExtension.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/websocketx/extensions/WebSocketClientExtension.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/websocketx/extensions/compression/PerFrameDeflateEncoder.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/websocketx/extensions/compression/PerFrameDeflateDecoder.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/websocketx/extensions/compression/PerMessageDeflateDecoder.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/websocketx/extensions/compression/PerMessageDeflateEncoder.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/spdy/SpdyHeaderBlockRawDecoder.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/spdy/SpdyFrameEncoder.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/websocketx/extensions/WebSocketExtension.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/websocketx/extensions/ServerCookieEncoder.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/websocketx/extensions/WebSocketClientExtensionHandler.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/websocketx/extensions/WebSocketServerExtensionHandler.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/websocketx/extensions/WebSocketClientExtension.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/websocketx/extensions/WebSocketServerExtension.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/websocketx/extensions/ServerCookieEncoder.java

No license file was found, but licenses were detected in source scan.

# The Netty Project licenses this file to you under the Apache License,
# version 2.0 (the "License"); you may not use this file except in compliance
# with the License. You may obtain a copy of the License at:
# distributed under the License is distributed on an "AS IS" BASIS, WITHOUT

Found in path(s):

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2013 The Netty Project
 * 
 * The Netty Project licenses this file to you under the Apache License, version
 * 2.0 (the "License"); you may not use this file except in compliance with the
 * License. You may obtain a copy of the License at:
 * 
 * https://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations under
 * the License.
 * */

Found in path(s):

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2019 The Netty Project
 * */
* The Netty Project licenses this file to you under the Apache License,*
* version 2.0 (the "License"); you may not use this file except in compliance*
* with the License. You may obtain a copy of the License at:
*
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software*
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT*
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the*
* License for the specific language governing permissions and limitations*
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/websocketx/WebSocketServerHandshakerFactory.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/websocketx/WebSocketCloseStatus.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/websocketx/extensions/WebSocketExtensionFilterProvider.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/websocketx/extensions/WebSocketExtensionFilter.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/websocketx/WebSocketServerHandshaker.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/websocketx/WebSocketDecoderConfig.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/websocketx/WebSocketServerHandshaker08.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/websocketx/WebSocketServerHandshaker00.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2016 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* 
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

found in path(s):
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/multipart/FileUploadUtil.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/HttpServerKeepAliveHandler.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/websocketx/WebSocketChunkedInput.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2014 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License, version 2.0 (the
* "License"); you may not use this file except in compliance with the License. You may obtain a
* copy of the License at:
* 
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

found in path(s):
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/HttpServerUpgradeHandler.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/HttpClientUpgradeHandler.java
No license file was found, but licenses were detected in source scan.

~ Copyright 2012 The Netty Project
~
~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~
~ https://www.apache.org/licenses/LICENSE-2.0
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.

Found in path(s):
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015 The Netty Project
 * *
 * The Netty Project licenses this file to you under the Apache License, version
 * 2.0 (the "License"); you may not use this file except in compliance with the
 * License. You may obtain a copy of the License at:
 * *
 * https://www.apache.org/licenses/LICENSE-2.0
 * *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2017 The Netty Project
 * *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/ReadOnlyHttpHeaders.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/websocketx/WebSocketScheme.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/HttpServerExpectContinueHandler.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2012 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

// (BSD License: https://www.opensource.org/licenses/bsd-license)
// All rights reserved.
// Redistribution and use in source and binary forms, with or
// * Redistributions of source code must retain the above
// copyright notice, this list of conditions and the
// following disclaimer.
// * Redistributions in binary form must reproduce the above
// following disclaimer in the documentation and/or other
// * Neither the name of the Webbit nor the names of

Found in path(s):
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/websocketx/WebSocket13FrameEncoder.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2020 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2013 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
No license file was found, but licenses were detected in source scan.
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2019 The Netty Project

...
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

*/

* Adaptation of https://bjoern.hoehrmann.de/utf-8/decoder/dfa/

* Copyright (c) 2008-2009 Bjoern Hoehrmann <bjoern@hoehrmann.de>

* Permission is hereby granted, free of charge, to any person obtaining a copy of this software
* and associated documentation files (the "Software"), to deal in the Software without restriction,
* including without limitation the rights to use, copy, modify, merge, publish, distribute,
* sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is
* furnished to do so, subject to the following conditions:
*
* The above copyright notice and this permission notice shall be included in all copies or
* substantial portions of the Software.
*
* THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
* IMPLIED, INCLUDING
* BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR
* PURPOSE AND
* NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE
* FOR ANY CLAIM,
* DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE,
* ARISING FROM,
* OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
* SOFTWARE.
*/

Found in path(s):
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
jar/io/netty/handler/codec/http/websocketx/Utf8Validator.java

No license file was found, but licenses were detected in source scan.

*/

* Copyright 2019 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
 *
* https://www.apache.org/licenses/LICENSE-2.0
 *
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

// (BSD License: https://www.opensource.org/licenses/bsd-license)
// All rights reserved.
// Redistribution and use in source and binary forms, with or
// * Redistributions of source code must retain the above
// * Redistributions of source code must retain the above
// * Redistributions in binary form must reproduce the above
// following disclaimer.
// * Redistributions in binary form must reproduce the above
// following disclaimer in the documentation and/or other
// * Neither the name of the Webbit nor the names of

Found in path(s):
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/websocketx/WebSocket08FrameDecoder.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/websocketx/WebSocket13FrameDecoder.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/websocketx/WebSocket07FrameDecoder.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2015 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
* https://www.apache.org/licenses/LICENSE-2.0
 *
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
jar/io/netty/handler/codec/http/CombinedHttpHeaders.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/rtsp/RtspEncoder.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/cookie/ClientCookieDecoder.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/cookie/CookieHeaderNames.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/cookie/ServerCookieEncoder.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/cookie/CombinedHttpHeaders.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/rtsp/RtspDecoder.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/cookie/CookieEncoder.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/cookie/package-info.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/EmptyHttpHeaders.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/cookie/CookieDecoder.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/cookie/CookieUtil.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/HttpScheme.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/HttpUtil.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/cookie/package-info.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/cookie/Cookie.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/HttpExpectationFailedEvent.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2021 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 */
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/HttpMessageDecoderResult.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/CompressionEncoderFactory.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/DefaultHttpMessage.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/multipart/AbstractHttpData.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/websocketx/PongWebSocketFrame.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/websocketx/BinaryWebSocketFrame.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/rtsp/RtspResponseEncoder.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/rtsp/RtspResponseStatuses.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/Header.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/multipart/MemoryFileUpload.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/multipart/AbstractHttpData.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/multipart/AbstractHttpData.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/multipart/AbstractHttpData.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2012 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/DefaultHttpMessage.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/multipart/AbstractHttpData.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/websocketx/PongWebSocketFrame.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/websocketx/BinaryWebSocketFrame.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/rtsp/RtspResponseEncoder.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/rtsp/RtspResponseStatuses.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/Header.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
  jar/io/netty/handler/codec/http/multipart/MemoryFileUpload.java
* /opt/cola/permits/1221850592_1635490079.45/0/netty-codec-http-4-1-69-final-sources-
1.5 spring-cloud-consul-core 3.0.4

1.5.1 Available under license:

No license file was found, but licenses were detected in source scan.

/ *
/ * Copyright 2013-2019 the original author or authors.
/ *
/ * Licensed under the Apache License, Version 2.0 (the "License");
/ * you may not use this file except in compliance with the License.
/ * You may obtain a copy of the License at
/ *
/ * https://www.apache.org/licenses/LICENSE-2.0
/ *
/ * Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1222421570_1635579136.39/0/spring-cloud-consul-core-3-0-4-sources-jar/org/springframework/cloud/consul/ConsulAutoConfiguration.java
* /opt/cola/permits/1222421570_1635579136.39/0/spring-cloud-consul-core-3-0-4-sources-jar/org/springframework/cloud/consul/ConditionalOnConsulEnabled.java
* /opt/cola/permits/1222421570_1635579136.39/0/spring-cloud-consul-core-3-0-4-sources-jar/org/springframework/cloud/consul/ConsulProperties.java
* /opt/cola/permits/1222421570_1635579136.39/0/spring-cloud-consul-core-3-0-4-sources-jar/org/springframework/cloud/consul/ConsulHealthIndicatorProperties.java
* /opt/cola/permits/1222421570_1635579136.39/0/spring-cloud-consul-core-3-0-4-sources-jar/org/springframework/cloud/consul/model/SerfStatusEnum.java
* /opt/cola/permits/1222421570_1635579136.39/0/spring-cloud-consul-core-3-0-4-sources-jar/org/springframework/cloud/consul/ConsulHealthIndicator.java
* /opt/cola/permits/1222421570_1635579136.39/0/spring-cloud-consul-core-3-0-4-sources-jar/org/springframework/cloud/consul/ConsulEndpoint.java
* /opt/cola/permits/1222421570_1635579136.39/0/spring-cloud-consul-core-3-0-4-sources-jar/org/springframework/cloud/consul/RetryProperties.java

No license file was found, but licenses were detected in source scan.

<name>Apache License, Version 2.0</name>
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

distributed under the License is distributed on an "AS IS" BASIS,

Found in path(s):
* /opt/cola/permits/1222421570_1635579136.39/0/spring-cloud-consul-core-3-0-4-sources-jar/META-INF/maven/org.springframework.cloud/spring-cloud-consul-core/pom.xml

1.6 async-http-client 2.11.0
1.6.1 Available under license :
No license file was found, but licenses were detected in source scan.

/**
* Copyright (c) 2017 AsyncHttpClient Project. All rights reserved.
* 
* This program is licensed to you under the Apache License Version 2.0,
* and you may not use this file except in compliance with the Apache License Version 2.0.
* You may obtain a copy of the Apache License Version 2.0 at http://www.apache.org/licenses/LICENSE-2.0.
* 
*/
* Unless required by applicable law or agreed to in writing,
* software distributed under the Apache License Version 2.0 is distributed on an
* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the Apache License Version 2.0 for the specific language governing permissions and limitations there under.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-jar/org/asynchnhttpclient/util/ThrowableUtil.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2010 Ning, Inc.
* *
* This program is licensed to you under the Apache License, version 2.0
* (the "License"); you may not use this file except in compliance with the
* License. You may obtain a copy of the License at:
* *
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-jar/org/asynchnhttpclient/AsyncCompletionHandlerBase.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-jar/org/asynchnhttpclient/AsyncCompletionHandler.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-jar/org/asynchnhttpclient/proxy/ProxyServer.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-jar/org/asynchnhttpclient/Request.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-jar/org/asynchnhttpclient/AsyncHttpClient.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2018 AsyncHttpClient Project. All rights reserved.
 *
 * This program is licensed to you under the Apache License Version 2.0,
 * and you may not use this file except in compliance with the Apache License Version 2.0.
 * You may obtain a copy of the Apache License Version 2.0 at
 *   http://www.apache.org/licenses/LICENSE-2.0.
 *
 * Unless required by applicable law or agreed to in writing,
 * software distributed under the Apache License Version 2.0 is distributed on an
 * "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the Apache License Version 2.0 for the specific language governing permissions and limitations there under.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-jar/org/asynchttpclient/netty/channel/PerHostConnectionSemaphore.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-jar/org/asynchttpclient/netty/channel/MaxConnectionSemaphore.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-jar/org/asynchttpclient/netty/channel/NoopConnectionSemaphore.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-jar/org/asynchttpclient/netty/channel/CombinedConnectionSemaphore.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-jar/org/asynchttpclient/oauth/ConsumerKey.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-jar/org/asynchttpclient/request/body/multipart/InputStreamPart.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-jar/org/asynchttpclient/netty/channel/ConnectionSemaphoreFactory.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-jar/org/asynchttpclient/netty/channel/DefaultConnectionSemaphoreFactory.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-jar/org/asynchttpclient/netty/channel/InfiniteSemaphore.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-jar/org/asynchttpclient/netty/channel/ConnectionSemaphore.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-jar/org/asynchttpclient/netty/channel/ConnectionSemaphore.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-jar/org/asynchttpclient/request/body/multipart/part/InputStreamMultipartPart.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-jar/org/asynchttpclient/proxy/ProxyType.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2015 AsyncHttpClient Project. All rights reserved.
 *
 * This program is licensed to you under the Apache License Version 2.0,
 * and you may not use this file except in compliance with the Apache License Version 2.0.
* You may obtain a copy of the Apache License Version 2.0 at
* http://www.apache.org/licenses/LICENSE-2.0.
*
* Unless required by applicable law or agreed to in writing,
* software distributed under the Apache License Version 2.0 is distributed on an
* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the Apache License Version 2.0 for the specific language governing permissions and limitations there under.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-
  jar/org/asynchttpclient/request/body/multipart/part/MultipartState.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-
  jar/org/asynchttpclient/resolver/RequestHostnameResolver.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-
  jar/org/asynchttpclient/request/body/multipart/MultipartBody.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-
  jar/org/asynchttpclient/netty/ssl/SslEngineFactoryBase.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-
  jar/org/asynchttpclient/request/body/multipart/part/StringMultipartPart.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-
  jar/org/asynchttpclient/netty/SimpleChannelFutureListener.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-
  jar/org/asynchttpclient/spnego/SpnegoEngineException.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-
  jar/org/asynchttpclient/netty/handler/intercept/ConnectSuccessInterceptor.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-
  jar/org/asynchttpclient/request/body/multipart/part/ByteArrayListPart.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-
  jar/org/asynchttpclient/netty/handler/intercept/Interceptors.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-
  jar/org/asynchttpclient/netty/handler/intercept/ResponseFiltersInterceptor.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-
  jar/org/asynchttpclient/util/Assertions.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-
  jar/org/asynchttpclient/request/body/multipart/part/MultipartPart.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-
  jar/org/asynchttpclient/netty/channel/ChannelState.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-
  jar/org/asynchttpclient/netty/request/body/NettyByteBufferBody.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-
  jar/org/asynchttpclient/request/body/multipart/part/MessageEndMultipartPart.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-
  jar/org/asynchttpclient/netty/ssl/JsseSslEngineFactory.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2010-2012 Sonatype, Inc. All rights reserved.
 *
 * This program is licensed to you under the Apache License Version 2.0,
 * and you may not use this file except in compliance with the Apache License Version 2.0.
 * You may obtain a copy of the Apache License Version 2.0 at http://www.apache.org/licenses/LICENSE-2.0.
 *
 * Unless required by applicable law or agreed to in writing,
 * software distributed under the Apache License Version 2.0 is distributed on an
 * "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the Apache License Version 2.0 for the specific language governing permissions and limitations there under.
 */
/*
* Licensed to the Apache Software Foundation (ASF) under one or more
* contributor license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright ownership.
* The ASF licenses this file to You under the Apache License, Version 2.0
* (the "License"); you may not use this file except in compliance with
* the License. You may obtain a copy of the License at
*
*      http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*===================================================================
* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
*
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-
  jar/org/asynchttpclient/spnego/SpnegoTokenGenerator.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2019 AsyncHttpClient Project. All rights reserved.
*
* This program is licensed to you under the Apache License Version 2.0,
* and you may not use this file except in compliance with the Apache License Version 2.0.
* You may obtain a copy of the Apache License Version 2.0 at
* http://www.apache.org/licenses/LICENSE-2.0.
*
* Unless required by applicable law or agreed to in writing,
* software distributed under the Apache License Version 2.0 is distributed on an
* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the Apache License Version 2.0 for the specific language governing permissions and limitations there under.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-
  jar/org/asynchttpclient/netty/channel/TransportFactory.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-
  jar/org/asynchttpclient/netty/channel/KQueueTransportFactory.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2014 AsyncHttpClient Project. All rights reserved.
 * 
 * This program is licensed to you under the Apache License Version 2.0,
 * and you may not use this file except in compliance with the Apache License Version 2.0.
 * You may obtain a copy of the Apache License Version 2.0 at
 * http://www.apache.org/licenses/LICENSE-2.0.
 * 
 * Unless required by applicable law or agreed to in writing,
 * software distributed under the Apache License Version 2.0 is distributed on an
 * "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the Apache License Version 2.0 for the specific language governing permissions and limitations there under.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-jar/org/asynchttpclient/netty/request/body/NettyBody.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-jar/org/asynchttpclient/ws/WebSocketUtils.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-jar/org/asynchttpclient/netty/request/body/BodyFileRegion.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-jar/org/asynchttpclient/netty/timeout/ReadTimeoutTimerTask.java
*/
No license file was found, but licenses were detected in source scan.

/*
 * Licensed to the Apache Software Foundation (ASF) under one or more
 * contributor license agreements. See the NOTICE file distributed with
 * this work for additional information regarding copyright ownership.
 * The ASF licenses this file to You under the Apache License, Version 2.0
 * (the "License"); you may not use this file except in compliance with
 * the License. You may obtain a copy of the License at
 *
 *    http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

/*
 * Copyright 2010 Ning, Inc.
 *
 * This program is licensed to you under the Apache License, version 2.0
 * (the "License"); you may not use this file except in compliance with the
 * License. You may obtain a copy of the License at:
 *
 *    http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/async-http-client-2-11-0-sources-jar/org/asynchttpclient/HttpResponseBodyPart.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/async-http-client-2-11-0-sources-jar/org/asynchttpclient/AsyncHandler.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/async-http-client-2-11-0-sources-jar/org/asynchttpclient/RequestBuilder.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/async-http-client-2-11-0-sources-jar/org/asynchttpclient/SignatureCalculator.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2016 AsyncHttpClient Project. All rights reserved.
 *
 * This program is licensed to you under the Apache License Version 2.0,
 * and you may not use this file except in compliance with the Apache License Version 2.0.
 * You may obtain a copy of the Apache License Version 2.0 at
 *   http://www.apache.org/licenses/LICENSE-2.0.
 *
 * Unless required by applicable law or agreed to in writing,
 * software distributed under the Apache License Version 2.0 is distributed on an
 * "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the Apache License Version 2.0 for the specific language governing permissions and limitations there under.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/async-http-client-2-11-0-sources-jar/org/asynchttpclient/netty/request/WriteCompleteListener.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/async-http-client-2-11-0-sources-jar/org/asynchttpclient/oauth/OAuthSignatureCalculator.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/async-http-client-2-11-0-sources-jar/org/asynchttpclient/netty/request/WriteListener.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/async-http-client-2-11-0-sources-jar/org/asynchttpclient/AsyncHttpClientState.java

No license file was found, but licenses were detected in source scan.

# This file maps Internet media types to unique file extension(s).
# Although created for httpd, this file is used by many software systems
# and has been placed in the public domain for unlimited redistribution.
#
# The table below contains both registered and (common) unregistered types.
# A type that has no unique extension can be ignored -- they are listed
# here to guide configurations toward known types and to make it easier to
# identify "new" types. File extensions are also commonly used to indicate
# content languages and encodings, so choose them carefully.
#
# Internet media types should be registered as described in RFC 4288.
# The registry is at <http://www.iana.org/assignments/media-types/>.
#
# MIME type (lowercased)Extensions
#
# application/1d-interleaved-parityfec
# application/3gpdash-qoe-report+xml
# application/3gpp-ims+xml
# application/a2l
# application/activemessage
# application/alto-costmap+json
# application/alto-costmapfilter+json
# application/alto-directory+json
# application/alto-endpointcost+json
# application/alto-endpointcostparams+json
# application/alto-endpointprop+json
# application/alto-endpointpropparams+json
# application/alto-error+json
# application/alto-networkmap+json
# application/alto-networkmapfilter+json
# application/aml
# application/andrew-insetez
# application/applefile
# application/applixwareaw
# application/atf
# application/atfx
# application/atom+xml
# application/atomcat+xml
# application/atomcat+xml
# application/atomdeleted+xml
# application/atomicmail
# application/atomsvc+xml
# application/atomsvc+xml
# application/axml
# application/auth-policy+xml
# application/bacnet-xdd+xml
# application/batch-smtplib
# application/beep+xml
# application/calendar+-json
# application/calendar+xml
# application/call-completion
# application/cals-1840
# application/ebor
# application/ccmp+xml
application/ccxml+xml
# application/cdf+xml
application/cdmi-capability
cdmia
application/cdmi-container
cdmic
application/cdmi-domain
cdmid
application/cdmi-object
cdmio
application/cdmi-queue
cdmiq
# application/cdni
# application/cea
# application/cea-2018+xml
# application/cellml+xml
# application/cfw
# application/cms
# application/cnrp+xml
# application/coap-group+json
# application/commonground
# application/conference-info+xml
# application/cpl+xml
# application/csrattrs
# application/csta+xml
# application/estdata+xml
# application/csvm+json
application/cu-seeme
cu
# application/cybercash
# application/davmount+xml
davmount
# application/dca-rft
davmount
# application/dcd
davmount
# application/dec-dx
davmount
# application/dialog-info+xml
davmount
# application/dicom
davmount
# application/dii
davmount
# application/dit
davmount
# application/dns
davmount
application/docbook+xml
dbk
# application/dskpp+xml
davmount
application/dssc+der
dssc
application/dssc+xml
dssc
# application/dvc
davmount
application/edimasciptecma
edimasciptecma
# application/edi-consent
# application/edi-x12
# application/edifact
# application/emergencycalldata.comment+xml
davmount
# application/emergencycalldata.deviceinfo+xml
davmount
application/scvp-cv-requestscq
application/scvp-cv-responsescscs
application/scvp-vp-requestspq
application/scvp-vp-responsespp
application/sdp
dp
# application/sep+xml
# application/sep-exi
# application/session-info
# application/set-payment
application/set-payment-initiationsetpay
# application/set-registration
application/set-registration-initiationsetreg
# application/sgml
# application/sgml-open-catalog
application/shf+xmlshf
# application/sieve
# application/simple-filter+xml
# application/simple-message-summary
# application/simplesymbolcontainer
# application/slate
# application/smil
application/smil+xmlsmilsmil
# application/smpte336m
# application/soap+fastinfoset
# application/soap+xml
application/sparql-query
application/sparql-results+xmlsrx
# application/spirits-event+xml
# application/sql
application/srgs
gram
application/srgs+xmlgrxml
application/sru+xmlsru
application/ssdl+xmlssdls SDL
application/ssml+xmlssml
# application/tamp-apex-update
# application/tamp-apex-update-confirm
# application/tamp-community-update
# application/tamp-community-update-confirm
# application/tamp-error
# application/tamp-sequence-adjust
# application/tamp-sequence-adjust-confirm
# application/tamp-status-query
# application/tamp-status-response
# application/tamp-update
# application/tamp-update-confirm
application/tei+xmltei teicorpus
application/thraud+xmltilf
# application/timestamp-query
# application/timestamp-reply
application/timestamped-datatsd
# application/ttml+xml
# application/tve-trigger
# application/ulpfec
# application/urc-grpsheet+xml
# application/urc-ressheet+xml
# application/urc-targetdesc+xml
# application/urc-uiosocketdesc+xml
# application/vcard+json
# application/vcard+xml
# application/vemmi
# application/vividence.scriptfile
# application/vnd.3gpp-prose+xml
# application/vnd.3gpp-prose-pc3ch+xml
# application/vnd.3gpp.access-transfer-events+xml
# application/vnd.3gpp.bsf+xml
# application/vnd.3gpp.mid-call+xml
application/vnd.3gpp.pic-bw-largeplb
application/vnd.3gpp.pic-bw-smallpsb
application/vnd.3gpp.pic-bw-varpvb
# application/vnd.3gpp.sms
# application/vnd.3gpp.srvc-ext+xml
# application/vnd.3gpp.srvc-info+xml
# application/vnd.3gpp.state-and-event-info+xml
# application/vnd.3gpp.ussd+xml
# application/vnd.3gpp2.bcmcsinfo+xml
# application/vnd.3gpp2.tcap
tcap
application/vnd.3m.post-it-notes pwn
application/vnd.accpac.simply.aso
application/vnd.acucobol acu
application/vnd.acucorp.acutc
application/vnd.adobe.air-application-installer-package+zipair
# application/vnd.adobe.flash.movie
application/vnd.adobe.formscentral.fcdtfcdt
application/vnd.adobe.fxpfxp fxp
# application/vnd.adobe.partial-upload
application/vnd.adobe.xdp+xmlxdp
application/vnd.adobe.xdfxdfdf
# application/vnd.aether.imp
# application/vnd.ah-barcode
application/vnd.ahead.spaceahead
application/vnd.airzip.filesecure.azfazf
application/vnd.airzip.filesecure.azsasz
application/vnd.amazon.ebookazw
application/vnd.americandynamics.accelc
Open Source Used In security_automation identity-guest-cs 101
# application/vnd.dvb.esgcontainer
# application/vnd.dvb.ipdcdftnotifaccess
# application/vnd.dvb.ipdcesgaccess
# application/vnd.dvb.ipdcesgaccess2
# application/vnd.dvb.ipdcesgpdd
# application/vnd.dvb.ipdcroaming
# application/vnd.dvb.iptv.alfec-base
# application/vnd.dvb.iptv.alfec-enhancement
# application/vnd.dvb.notif-aggregate-root+xml
# application/vnd.dvb.notif-container+xml
# application/vnd.dvb.notif-generic+xml
# application/vnd.dvb.notif-ia-msglist+xml
# application/vnd.dvb.notif-ia-registration-request+xml
# application/vnd.dvb.notif-ia-registration-response+xml
# application/vnd.dvb.notif-init+xml
# application/vnd.dvb.pfr
application/vnd.dvb.servicesvc
# application/vnd.dvb.dxr
application/vnd.dynageogeo
# application/vnd.dvb.dzr
# application/vnd.dvb.ecdis-update
application/vnd.ecowin.chartmag
# application/vnd.ecowin.filerequest
# application/vnd.ecowin.fileupdate
# application/vnd.ecowin.series
# application/vnd.ecowin.seriesrequest
# application/vnd.ecowin.seriesupdate
# application/vnd.ecowin.emclient.accessrequest+xml
application/vnd.enlivenxml
# application/vnd.enphase.envoy
# application/vnd.epis.data+xml
application/vnd.epson.esf
application/vnd.epson.msf
application/vnd.epson.quickanimeqam
application/vnd.epson.saltssl
application/vnd.epson.ssfsf
# application/vnd.ericsson.quickcall
application/vnd.eszigno3+xml+es3 et3
# application/vnd.etsi.aoc+xml
# application/vnd.etsi.asic-e+zip
# application/vnd.etsi.asic-s+zip
# application/vnd.etsi.cug+xml
# application/vnd.etsi.iptvcommand+xml
# application/vnd.etsi.iptvdiscovery+xml
# application/vnd.etsi.iptvprofile+xml
# application/vnd.etsi.iptvsad-bc+xml
# application/vnd.etsi.iptvsad-cod+xml
Open Source Used In security_automation identity-guest-cs 111
# application/vnd.sealed.xls
# application/vnd.sealedmedia.softseal.html
# application/vnd.sealedmedia.softseal.pdf
application/vnd.seemail see
application/vnd.sema sema
application/vnd.semd semd
application/vnd.semf semf
application/vnd.shana.informed.formdata ifm
application/vnd.shana.informed.formtemplate itp
application/vnd.shana.informed.interchange if
application/vnd.shana.informed.package ipk
application/vnd.simtech-mindmappertwd twds
# application/vnd.siren+json
application/vnd.smaf mmf
# application/vnd.smart.notebook
application/vnd.smart.teacherteacher
# application/vnd.software602.filler.form+xml
# application/vnd.software602.filler.form-xml-zip
application/vnd.solent.sdm+xml xmlsd d km sdk d
application/vnd.spotfire.dxp d xp
tapplication/vnd.spotfire.sfs fs
tapplication/vnd.sss-cod
tapplication/vnd.sss-dtf
tapplication/vnd.sss-ntf
application/vnd.stardivision.cals dc
application/vnd.stardivision.draw d sa
tapplication/vnd.stardivision.impress ss d d
application/vnd.stardivision.mathsmf
application/vnd.stardivision.writers dw vor
application/vnd.stardivision.writer-globals gl
application/vnd.stepmania.packagesm zip
application/vnd.stepmania.stepchart sm
# application/vnd.street-stream
# application/vnd.sun.wadl+xml
application/vnd.sun.xml.cals xc
tapplication/vnd.sun.xml.calt em plate stc
tapplication/vnd.sun.xml.draw x d
tapplication/vnd.sun.xml.draw.template std
tapplication/vnd.sun.xml.impress ss xi
application/vnd.sun.xml.impress.template sti
application/vnd.sun.xml.mathsx m
application/vnd.sun.xml.writers xw
application/vnd.sun.xml.writer.global s x g
application/vnd.sun.xml.writer.template stw
application/vnd.sus-calendars susp
application/vnd.svd svd
# application/vnd.swiftview-ics
application/vnd.symbian.install sis six
audio/vnd.nuera.ecelp7470ecelp7470
audio/vnd.nuera.ecelp9600ecelp9600
# audio/vnd.octel.sbc
# audio/vnd.qcelp
# audio/vnd.rhetorex.32kadpcm
audio/vnd.riprip
# audio/vnd.sealedmedia.softseal.mpeg
# audio/vnd.vmx.cvsd
# audio/vorbis
# audio/vorbis-config
audio/webmweba
audio/x-aacaac
audio/x-aiff aiff aifc
audio/x-cafcaf
audio/x-flacflac
audio/x-matroskamka
audio/x-mpegurlm3u
audio/x-ms-waxwax
audio/x-ms-wmawma
audio/x-pn-realaudiooram ra
audio/x-pn-realaudio-pluginrpm
# audio/x-tta
audio/x-wavwav
audio/xmxml
chemical/x-cdxcdx
chemical/x-cicif
chemical/x-cmdfcmdf
chemical/x-cmleml
chemical/x-csmlcsml
# chemical/x-pdb
chemical/x-xyzxyz
image/bmpbmp
image/cgmkgm
# image/example
# image/fits
image/g3faxg3
image/gifgif
image/iefief
# image/jp2
image/jpegjpeg jpg jpe
# image/jpm
# image/jpx
image/ktxktx
# image/naplps
image/pngpng
image/prs.btifbtif
# image/prs.pti
# image/pwg-raster
# text/rtp-enc-aescm128
# text/rtploopback
# text/rtx
text/sgmlsgml sgml
# text/140
text/tab-separated-values
# text/t140
text/troff tr roff man me ms
text/turtle
# text/ulpfec
text/uri-listuri uris urls
text/vcardvcard
# text/vnd.a
text/vnd.curlcurl
text/vnd.curl.dcurlcurl
text/vnd.curl.mcurlmcurl
text/vnd.curl.scurlscurl
# text/vnd.debian.copyright
# text/vnd.dmclientscript
text/vnd.dvb.subtitlesub
# text/vnd.esmertec.theme-descriptor
text/vnd.flyfly
text/vnd.fmi.flexstorflx
text/vnd.graphvizgv
text/vnd.in3d.3dml3dml
text/vnd.in3d.spotspot
# text/vnd.iptc.newsml
# text/vnd.iptc.nif
# text/vnd.latex-z
# text/vnd.motorola.reflex
# text/vnd.ms-mediapackage
# text/vnd.net2phone.commcenter.command
# text/vnd.radisys.msml-basic-layout
# text/vnd.si.uricatalogue
text/vnd.sun.j2me.app-descriptorjad
# text/vnd.trolltech.linguist
# text/vnd.wap.si
# text/vnd.wap.sl
text/vnd.wap.wmlwml
text/vnd.wap.wmlscriptwmls
text/x-asmsasm
text/x-cc ccc xcc cpp h hh dic
text/x-fortranf for f77 f90
text/x-java-sourcejava
text/x-nfinfo
# text/x-opmloptml
# text/x-pascalpas
# text/x-setexttex
text/x-sfvsfv

text/x-uuencode

text/x-vcalendar

text/x-vcard

# text/xml

# text/xml-external-parsed-entity

video/1d-interleaved-parityfec

video/3gp

# video/3gpp-tt

video/3gpp23g2

# video/bmpeg

# video/bt656

# video/celb

# video/dv

# video/encaprtp

# video/example

video/h261h261

video/h263h263

# video/h263-1998

# video/h263-2000

video/h264h264

# video/h264-rcdo

# video/h264-svc

# video/h265

# video/iso.segment

video/jpegjgv

# video/jpeg2000

video/jpmjpm jpgm

video/mj2mj2 mj2p

# video/mp1s

# video/mp2p

# video/mp2t

video/mp4mp4 mp4v mpg4

# video/mp4v-es

video/mpegmpeg mpg m1v m2v

# video/mpeg4-generic

# video/mpv

# video/nv

video/oggoggv

# video/parityfec

# video/pointer

video/quicktimeqt mov

# video/raptorfec

# video/raw

# video/rtp-enc-aescm128

# video/rtploopback

# video/rtx

# video/smpte292m
video/x-ms-wmxwmx
video/x-ms-wvxwvx
video/x-msvideo
video/x-sgi-movie
video/x-smv
x-conference/x-cooltalkice

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-jar/org/asynchttpclient/request/body/multipart/ahc-mime.types
No license file was found, but licenses were detected in source scan.

/**
 * Copyright (c) 2014 AsyncHttpClient Project. All rights reserved.
 *
 * This program is licensed to you under the Apache License Version 2.0,
 * and you may not use this file except in compliance with the Apache License Version 2.0.
 * You may obtain a copy of the Apache License Version 2.0 at http://www.apache.org/licenses/LICENSE-2.0.
 *
 * Unless required by applicable law or agreed to in writing,
 * software distributed under the Apache License Version 2.0 is distributed on an
 * "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the Apache License Version 2.0 for the specific language governing permissions and limitations there under.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-jar/org/asynchttpclient/request/body/multipart/Part.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-jar/org/asynchttpclient/netty/LazyResponseBodyPart.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-jar/org/asynchttpclient/request/body/multipart/PartBase.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-jar/org/asynchttpclient/request/body/multipart/part/PartVisitor.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-jar/org/asynchttpclient/BoundRequestBuilder.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-jar/org/asynchttpclient/util/UriEncoder.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-jar/org/asynchttpclient/netty/OnLastHttpContentCallback.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-jar/org/asynchttpclient/Param.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-jar/org/asynchttpclient/request/body/multipart/ByteArrayPart.java
/*
 * Copyright (c) 2017 AsyncHttpClient Project. All rights reserved.
 *
 * This program is licensed to you under the Apache License Version 2.0,
 * and you may not use this file except in compliance with the Apache License Version 2.0.
 * You may obtain a copy of the Apache License Version 2.0 at
 *     http://www.apache.org/licenses/LICENSE-2.0.
 *
 * Unless required by applicable law or agreed to in writing,
 * software distributed under the Apache License Version 2.0 is distributed on an
 * "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the Apache License Version 2.0 for the specific language governing permissions and limitations there under.
 */

No license file was found, but licenses were detected in source scan.
jar/org/asynchttpclient/cookie/CookieStore.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2010-2012 Sonatype, Inc. All rights reserved.
*
* This program is licensed to you under the Apache License Version 2.0,
* and you may not use this file except in compliance with the Apache License Version 2.0.
* You may obtain a copy of the Apache License Version 2.0 at http://www.apache.org/licenses/LICENSE-2.0.
*
* Unless required by applicable law or agreed to in writing,
* software distributed under the Apache License Version 2.0 is distributed on an
* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the Apache License Version 2.0 for the specific language governing permissions and limitations there under.
*/
/*
*===================================================================
* Licensed to the Apache Software Foundation (ASF) under one or more
* contributor license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright ownership.
* The ASF licenses this file to You under the Apache License, Version 2.0
* (the "License"); you may not use this file except in compliance with
* the License. You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*===================================================================
*
* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/async-http-client-2-11-0-sources-
jar/org/asynchttpclient/spnego/SpnegoEngine.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2010 Ning, Inc.
* */
* This program is licensed to you under the Apache License, version 2.0
* (the "License"); you may not use this file except in compliance with the
* License. You may obtain a copy of the License at:
* 
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/
/
* Copyright (C) 2007 Google Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-
  jar/org/asynchttpclient/ListenableFuture.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2015 AsyncHttpClient Project. All rights reserved.
* 
* This program is licensed to you under the Apache License Version 2.0,
* and you may not use this file except in compliance with the Apache License Version 2.0.
* You may obtain a copy of the Apache License Version 2.0 at http://www.apache.org/licenses/LICENSE-2.0.
* 
* Unless required by applicable law or agreed to in writing,
* software distributed under the Apache License Version 2.0 is distributed on an
* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the Apache License Version 2.0 for the specific language governing permissions and limitations there under.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2010-2013 Sonatype, Inc. All rights reserved.
 * 
 * This program is licensed to you under the Apache License Version 2.0,
 * and you may not use this file except in compliance with the Apache License Version 2.0.
 * You may obtain a copy of the Apache License Version 2.0 at http://www.apache.org/licenses/LICENSE-2.0.
 * 
 * Unless required by applicable law or agreed to in writing,
 * software distributed under the Apache License Version 2.0 is distributed on an
 * "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the Apache License Version 2.0 for the specific language governing permissions and limitations there under.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-jar/org/asynchttpclient/util/AuthenticatorUtils.java
No license file was found, but licenses were detected in source scan.

/*
 * =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
 * Licensed to the Apache Software Foundation (ASF) under one
 * or more contributor license agreements. See the NOTICE file
 * distributed with this work for additional information
 * regarding copyright ownership. The ASF licenses this file
 * to you under the Apache License, Version 2.0 (the
 * "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at
 */
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing,
* software distributed under the License is distributed on an
* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
* KIND, either express or implied. See the License for the
* specific language governing permissions and limitations
* under the License.
* ============================== http://www.apache.org/licenses/LICENSE-2.0 ===========
*
* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
*
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-
  jar/org/asynchttpclient/ntlm/NtlmEngine.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2010-2013 Ning, Inc.
*
* This program is licensed to you under the Apache License, version 2.0
* (the "License"); you may not use this file except in compliance with the
* License. You may obtain a copy of the License at:
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-
  jar/org/asynchttpclient/RequestBuilderBase.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2010-2012 Sonatype, Inc. All rights reserved.
*
* This program is licensed to you under the Apache License Version 2.0,
* and you may not use this file except in compliance with the Apache License Version 2.0.
* You may obtain a copy of the Apache License Version 2.0 at http://www.apache.org/licenses/LICENSE-2.0.
* 
* Unless required by applicable law or agreed to in writing,
* software distributed under the Apache License Version 2.0 is distributed on an
* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the Apache License Version 2.0 for the specific language governing permissions and limitations there under.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-jar/org/asynchnpcient/handler/ProgressAsyncHandler.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-jar/org/asynchnpcient/handler/TransferCompletionHandler.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-jar/org/asynchnpcient/util/MiscUtils.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-jar/org/asynchnpcient/handler/resumable/PropertiesBasedResumableProcessor.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-jar/org/asynchnpcient/ws/WebSocketListener.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-jar/org/asynchnpcient/webdav/WebDavCompletionHandlerBase.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-jar/org/asynchnpcient/request/body/generator/ByteArrayBodyGenerator.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-jar/org/asynchnpcient/request/body/Body.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-jar/org/asynchnpcient/webdav/WebDavCompletionHandler.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-jar/org/asynchnpcient/request/body/generator/ByteArrayBodyGenerator.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-jar/org/asynchnpcient/request/body/Body.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-jar/org/asynchnhttpclient/request/body/generator/BodyGenerator.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-jar/org/asynchnhttpclient/util/HttpUtils.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-jar/org/asynchnhttpclient/Realm.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-jar/org/asynchnhttpclient/filter/ThrottleRequestFilter.java
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-jar/org/asynchnhttpclient/request/body/generator/InputStreamBodyGenerator.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2010-2013 Ning, Inc.
 *
 * This program is licensed to you under the Apache License, version 2.0
 * (the "License"); you may not use this file except in compliance with the
 * License. You may obtain a copy of the License at:
 * *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 *
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-jar/org/asynchnhttpclient/Realm.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2017 AsyncHttpClient Project. All rights reserved.
 *
 * This program is licensed to you under the Apache License Version 2.0,
 * and you may not use this file except in compliance with the Apache License Version 2.0.
 * You may obtain a copy of the Apache License Version 2.0 at
 * http://www.apache.org/licenses/LICENSE-2.0.
 */
* Unless required by applicable law or agreed to in writing,
* software distributed under the Apache License Version 2.0 is distributed on an
* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the Apache License Version 2.0 for the specific language governing permissions and limitations there under.
*/
/**
* Value class used for OAuth tokens (request secret, access secret);
* simple container with two parts, public id part ("key") and
* confidential ("secret") part.
*/

```
Found in path(s):

* /opt/ws_local/PERMITS_SQL/1030071614_1591326199.5/0/async-http-client-2-11-0-sources-jar/org/asynchttpclient/oauth/RequestToken.java
```

1.7 postgresql-jdbc 42.2.24

1.7.1 Available under license:

Copyright 2017, OnGres.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

```
THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```
this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice,
   this list of conditions and the following disclaimer in the documentation
   and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE
POSSIBILITY OF SUCH DAMAGE.

Additional License files can be found in the 'licenses' folder located in the same directory as the LICENSE file (i.e.
this file)

- Software produced outside the ASF which is available under other licenses (not Apache-2.0)

BSD-2-Clause
* com.ongres.scram:client:2.1
* com.ongres.scram:common:2.1
* com.ongres.stringprep:saslprep:1.1
* com.ongres.stringprep:stringprep:1.1
Copyright 2019, OnGres.

Redistribution and use in source and binary forms, with or without modification,
are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this
   list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice,
   this list of conditions and the following disclaimer in the documentation and/or
   other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR
ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON
ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
1.8 spring-beans 5.3.12
1.8.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2018 the original author or authors.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
*/

---

Open Source Used In security_automation identity-guest-cs 140
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2010 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *      https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-
jar/org/springframework/beans/factory/support/BeanDefinitionRegistryPostProcessor.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2011 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *      https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2007 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *   https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2016 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *   https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
  * limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178460_1635318352_56/0/spring-beans-5-3-12-sources-
  jar/org/springframework/beans/propertyeditors/ReaderEditor.java
* /opt/cola/permits/1220178460_1635318352_56/0/spring-beans-5-3-12-sources-
  jar/org/springframework/beans/factory/NamedBean.java
* /opt/cola/permits/1220178460_1635318352_56/0/spring-beans-5-3-12-sources-
  jar/org/springframework/beans/factory/config/SmartInstantiationAwareBeanPostProcessor.java
* /opt/cola/permits/1220178460_1635318352_56/0/spring-beans-5-3-12-sources-
  jar/org/springframework/beans/factory/config/Scope.java
* /opt/cola/permits/1220178460_1635318352_56/0/spring-beans-5-3-12-sources-
  jar/org/springframework/beans/propertyeditors/FileEditor.java
* /opt/cola/permits/1220178460_1635318352_56/0/spring-beans-5-3-12-sources-
  jar/org/springframework/beans/factory/BeanInitializationException.java
* /opt/cola/permits/1220178460_1635318352_56/0/spring-beans-5-3-12-sources-
  jar/org/springframework/beans/factory/xml/UtilNamespaceHandler.java
* /opt/cola/permits/1220178460_1635318352_56/0/spring-beans-5-3-12-sources-
  jar/org/springframework/beans/factory/support/ImplicitlyAppearedSingletonException.java
* /opt/cola/permits/1220178460_1635318352_56/0/spring-beans-5-3-12-sources-
  jar/org/springframework/beans/factory/parsing/SourceExtractor.java
* /opt/cola/permits/1220178460_1635318352_56/0/spring-beans-5-3-12-sources-
  jar/org/springframework/beans/propertyeditors/URLEditor.java
* /opt/cola/permits/1220178460_1635318352_56/0/spring-beans-5-3-12-sources-
  jar/org/springframework/beans/support/ResourceIdEditor.java
* /opt/cola/permits/1220178460_1635318352_56/0/spring-beans-5-3-12-sources-
  jar/org/springframework/beans/factory/xml/NamespaceHandlerResolver.java
* /opt/cola/permits/1220178460_1635318352_56/0/spring-beans-5-3-12-sources-
  jar/org/springframework/beans/factory/config/BeanDefinitionCustomizer.java

No license file was found, but licenses were detected in source scan.

/

* Copyright 2002-2020 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* https://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-jar/org/springframework/beans/BeanWrapperImpl.java
* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-jar/org/springframework/beans/factory/support/BeanDefinitionReaderUtils.java
* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-jar/org/springframework/beans/factory/support/PropertiesBeanDefinitionReader.java
* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-jar/org/springframework/beans/factory/support/SimpleAutowireCandidateResolver.java
* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-jar/org/springframework/beans/factory/support/AutowireUtils.java
* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-jar/org/springframework/beans/factory/ListableBeanFactory.java
* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-jar/org/springframework/beans/factory/FactoryBean.java
* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-jar/org/springframework/beans/factory/CachedIntrospectionResults.java
* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-jar/org/springframework/beans/factory/config/ConfigurableBeanFactory.java
* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-jar/org/springframework/beans/factory/xml/XmlBeanDefinitionReader.java
* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-jar/org/springframework/beans/factory/propertyeditors/URIEditor.java
* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-jar/org/springframework/beans/factory/annotation/InjectionMetadata.java
* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-jar/org/springframework/beans/factory/support/BeanDefinitionDefaults.java
* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-jar/org/springframework/beans/factory/config/InstantiationAwareBeanPostProcessorAdapter.java
* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-jar/org/springframework/beans/factory/parsing/ParseState.java
* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-jar/org/springframework/beans/factory/parsing/QualifierEntry.java
* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-jar/org/springframework/beans/factory/annotation/QualifierAnnotationAutowireCandidateResolver.java
* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-jar/org/springframework/beans/factory/support/DefaultListableBeanFactory.java
* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-jar/org/springframework/beans/factory/support/AutowireCandidateResolver.java
* No license file was found, but licenses were detected in source scan.

/*
* Copyright 2002-2015 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
*   https://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-
  jar/org/springframework/beans/factory/xml/BeanDefinitionDocumentReader.java
* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-
  jar/org/springframework/beans/factory/config/SingletonBeanRegistry.java
* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-
  jar/org/springframework/beans/factory/xml/AbstractSingleBeanDefinitionParser.java
* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-
  jar/org/springframework/beans/BeanWrapper.java
* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-
  jar/org/springframework/beans/factory/config/EmbeddedValueResolver.java
* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-
  jar/org/springframework/beans/factory/xml/XmlReaderContext.java
* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-
  jar/org/springframework/beans/factory/xml/NamespaceHandlerSupport.java

/*
* Copyright 2002-2017 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-
  jar/org/springframework/beans/factory/xml/AbstractSingleBeanDefinitionParser.java
* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-
  jar/org/springframework/beans/BeanWrapper.java
* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-
  jar/org/springframework/beans/factory/config/EmbeddedValueResolver.java
* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-
  jar/org/springframework/beans/factory/xml/XmlReaderContext.java
* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-
  jar/org/springframework/beans/factory/xml/NamespaceHandlerSupport.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2021 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *      https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 */
* See the License for the specific language governing permissions and
  * limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-
  jar/org/springframework/beans/factory/support/ConstructorResolver.java
* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-
  jar/org/springframework/beans/factory/xml/ParserContext.java
* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-
  jar/org/springframework/beans/propertyeditors/PathEditor.java
* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-
  jar/org/springframework/beans/factory/support/LookupOverride.java
* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-
  jar/org/springframework/beans/factory/groovy/GroovyBeanDefinitionReader.java
* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-
  jar/org/springframework/beans/factory/annotation/RequiredAnnotationBeanPostProcessor.java
* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-
  jar/org/springframework/beans/factory/support/RootBeanDefinition.java
* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-
  jar/org/springframework/beans/propertyeditors/LocaleEditor.java
* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-
  jar/org/springframework/beans/factory/config/PlaceholderConfigurerSupport.java
* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-
  jar/org/springframework/beans/factory/support/MethodReplacer.java
* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-
  jar/org/springframework/beans/support/PropertyComparator.java
* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-
  jar/org/springframework/beans/GenericTypeAwarePropertyDescriptor.java
* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-
  jar/org/springframework/beans/factory/config/PlaceholderConfigurerSupport.java
* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-
  jar/org/springframework/beans/support/PropertyComparator.java
* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-
  jar/org/springframework/beans/GenericTypeAwarePropertyDescriptor.java
* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-
/*
 * Copyright 2002-2019 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *      https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

No license file was found, but licenses were detected in source scan.
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2012 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *      https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-jar/org/springframework/beans/factory/config/YamlPropertiesFactoryBean.java
* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-jar/org/springframework/beans/factory/annotation/InitDestroyAnnotationBeanPostProcessor.java
* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-jar/org/springframework/beans/factory/config/AutowireCapableBeanFactory.java
* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-jar/org/springframework/beans/factory/support/GenericBeanDefinition.java
* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-jar/org/springframework/beans/factory/xml/BeansDtdResolver.java
* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-jar/org/springframework/beans/factory/xml/DocumentLoader.java
* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-jar/org/springframework/beans/support/MutableSortDefinition.java
* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-jar/org/springframework/beans/factory/parsing/ComponentDefinition.java
* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-jar/org/springframework/beans/factory/parsing/ConstructorArgumentEntry.java
* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-jar/org/springframework/beans/factory/config/PropertyResourceConfigurer.java
*/
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2013 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-jar/org/springframework/beans/propertyeditors/CharsetEditor.java
* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-jar/org/springframework/beans/PropertyEditorRegistrar.java
* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-jar/org/springframework/beans/factory/BeanCurrentlyInCreationException.java
* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-jar/org/springframework/beans/factory/xml/XmlBeanDefinitionStoreException.java
* /opt/cola/permits/1220178460_1635318352.56/0/spring-beans-5-3-12-sources-jar/org/springframework/beans/factory/xml/SimplePropertyNamespaceHandler.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2014 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * *
 * https://www.apache.org/licenses/LICENSE-2.0
 */
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178460_1635318352.56_0/spring-beans-5-3-12-sources-
  jar/org/springframework/beans/factory/SmartInitializingSingleton.java
* /opt/cola/permits/1220178460_1635318352.56_0/spring-beans-5-3-12-sources-
  jar/org/springframework/beans/factory/xml/AbstractSimpleBeanDefinitionParser.java
* /opt/cola/permits/1220178460_1635318352.56_0/spring-beans-5-3-12-sources-
  jar/org/springframework/beans/factory/annotation/AnnotatedBeanDefinition.java
* /opt/cola/permits/1220178460_1635318352.56_0/spring-beans-5-3-12-sources-
  jar/org/springframework/beans/NotReadablePropertyException.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2002-2009 the original author or authors.
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178460_1635318352.56_0/spring-beans-5-3-12-sources-
  jar/org/springframework/beans/factory/annotation/Autowire.java

1.9 fast-infoset 1.2.13

1.9.1 Available under license:
No license file was found, but licenses were detected in source scan.

/*
* DO NOT ALTER OR REMOVE COPYRIGHT NOTICES OR THIS HEADER.
* */
* Copyright (c) 2004-2011 Oracle and/or its affiliates. All rights reserved.
* 
* Oracle licenses this file to You under the Apache License, Version 2.0
* (the "License"); you may not use this file except in compliance with
* the License. You may obtain a copy of the License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
/*
* This code is subject to the freebxml License, Version 1.1
* 
* Copyright (c) 2001 - 2005 freebxml.org. All rights reserved.
* 
* $Header:
/zpool01/javanet/scm/svn/tmp/cvs2svn/fi/FastInfoset/src/com/sun/xml/fastinfoset/AbstractResourceBundle.java,v
1.3.2.4 2009-05-13 08:53:01 oleksiys Exp $
*/

Found in path(s):
* /opt/cola/permits/1000966784_1646171595.15/0/fastinfoset-1-2-13-sources-
jar/com/sun/xml/fastinfoset/AbstractResourceBundle.java
No license file was found, but licenses were detected in source scan.

/*
* DO NOT ALTER OR REMOVE COPYRIGHT NOTICES OR THIS HEADER.
* 
* Copyright (c) 2004-2013 Oracle and/or its affiliates. All rights reserved.
* 
* Oracle licenses this file to You under the Apache License, Version 2.0
* (the "License"); you may not use this file except in compliance with
* the License. You may obtain a copy of the License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
Found in path(s):
* /opt/cola/permits/1000966784_1646171595.15/0/fastinfoset-1-2-13-sources-jar/com/sun/xml/fastinfoset/Decoder.java
* /opt/cola/permits/1000966784_1646171595.15/0/fastinfoset-1-2-13-sources-jar/com/sun/xml/fastinfoset/sax/AttributesHolder.java
* /opt/cola/permits/1000966784_1646171595.15/0/fastinfoset-1-2-13-sources-jar/com/sun/xml/fastinfoset/algorithm/BASE64EncodingAlgorithm.java
* /opt/cola/permits/1000966784_1646171595.15/0/fastinfoset-1-2-13-sources-jar/com/sun/xml/fastinfoset/algorithm/HexadecimalEncodingAlgorithm.java
* /opt/cola/permits/1000966784_1646171595.15/0/fastinfoset-1-2-13-sources-jar/com/sun/xml/fastinfoset/sax/SAXDocumentParser.java
* /opt/cola/permits/1000966784_1646171595.15/0/fastinfoset-1-2-13-sources-jar/com/sun/xml/fastinfoset/algorithm/BuiltInEncodingAlgorithm.java
* /opt/cola/permits/1000966784_1646171595.15/0/fastinfoset-1-2-13-sources-jar/com/sun/xml/fastinfoset/algorithm/FloatEncodingAlgorithm.java
* /opt/cola/permits/1000966784_1646171595.15/0/fastinfoset-1-2-13-sources-jar/com/sun/xml/fastinfoset/stax/StAXDocumentParser.java
* /opt/cola/permits/1000966784_1646171595.15/0/fastinfoset-1-2-13-sources-jar/com/sun/xml/fastinfoset/algorithm/BooleanEncodingAlgorithm.java
No license file was found, but licenses were detected in source scan.

/*
 * DO NOT ALTER OR REMOVE COPYRIGHT NOTICES OR THIS HEADER.
 *
 * Copyright (c) 2005-2011 Oracle and/or its affiliates. All rights reserved.
 *
 * Oracle licenses this file to You under the Apache License, Version 2.0
 * (the "License"); you may not use this file except in compliance with
 * the License. You may obtain a copy of the License at
 *
 *      http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1000966784_1646171595.15/0/fastinfoset-1-2-13-sources-jar/com/sun/xml/fastinfoset/stax/events/StAXFilteredEvent.java
No license file was found, but licenses were detected in source scan.

/*
 * DO NOT ALTER OR REMOVE COPYRIGHT NOTICES OR THIS HEADER.
 *
 * Copyright (c) 2004-2011 Oracle and/or its affiliates. All rights reserved.
 */
* Oracle licenses this file to You under the Apache License, Version 2.0
* (the "License"); you may not use this file except in compliance with
* the License. You may obtain a copy of the License at
* *
* http://www.apache.org/licenses/LICENSE-2.0
* *
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
/*
* The Apache Software License, Version 1.1
* *
* Copyright (c) 1999-2004 The Apache Software Foundation. All rights
* reserved.
* *
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* *
* 1. Redistributions of source code must retain the above copyright
*    notice, this list of conditions and the following disclaimer.
* *
* 2. Redistributions in binary form must reproduce the above copyright
*    notice, this list of conditions and the following disclaimer in
*    the documentation and/or other materials provided with the
*    distribution.
* *
* 3. The end-user documentation included with the redistribution,
*    if any, must include the following acknowledgment:
*    "This product includes software developed by the
*    Apache Software Foundation (http://www.apache.org/)."
* Alternately, this acknowledgment may appear in the software itself,
* if and wherever such third-party acknowledgments normally appear.
* *
* 4. The names "Xerces" and "Apache Software Foundation" must
*    not be used to endorse or promote products derived from this
*    software without prior written permission. For written
*    permission, please contact apache@apache.org.
* *
* 5. Products derived from this software may not be called "Apache",
*    nor may "Apache" appear in their name, without prior written
*    permission of the Apache Software Foundation.
*
/* THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESSED OR IMPLIED
 * WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
 * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
 * DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
 * ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
 * SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
 * LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
 * USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
 * ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
 * OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
 * OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
 * SUCH DAMAGE.
 * ====================================================================
 *
 * This software consists of voluntary contributions made by many
 * individuals on behalf of the Apache Software Foundation and was
 * originally based on software copyright (c) 1999, International
 * Business Machines, Inc., http://www.apache.org. For more
 * information on the Apache Software Foundation, please see
 */

Found in path(s):
* /opt/cola/permits/1000966784_1646171595.15/0/fastinfoset-1-2-13-sources-
  jar/com/sun/xml/fastinfoset/org/apache/xerces/util/XMLChar.java
No license file was found, but licenses were detected in source scan.

/*
 * DO NOT ALTER OR REMOVE COPYRIGHT NOTICES OR THIS HEADER.
 *
 * Copyright (c) 2004-2011 Oracle and/or its affiliates. All rights reserved.
 *
 * Oracle licenses this file to You under the Apache License, Version 2.0
 * (the "License"); you may not use this file except in compliance with
 * the License. You may obtain a copy of the License at
 *
 *     http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1000966784_1646171595.15/0/fastinfoset-1-2-13-sources-
  jar/org/jvnet/fastinfoset/FastInfoSetSerializer.java
No license file was found, but licenses were detected in source scan.

# Copyright (c) 2012 Oracle and/or its affiliates. All rights reserved.
# Oracle licenses this file to You under the Apache License, Version 2.0
# (the "License"); you may not use this file except in compliance with
# the License. You may obtain a copy of the License at
# http://www.apache.org/licenses/LICENSE-2.0
# distributed under the License is distributed on an "AS IS" BASIS,

Found in path(s):
* /opt/cola/permits/100966784_1646171595.15/0/fastinfoset-1-2-13-sources-
  jar/com/sun/xml/fastinfoset/resources/ResourceBundle.properties
No license file was found, but licenses were detected in source scan.

/*
* DO NOT ALTER OR REMOVE COPYRIGHT NOTICES OR THIS HEADER.
* 
* Copyright (c) 2004-2012 Oracle and/or its affiliates. All rights reserved.
* 
* Oracle licenses this file to You under the Apache License, Version 2.0
* (the "License"); you may not use this file except in compliance with
* the License. You may obtain a copy of the License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/100966784_1646171595.15/0/fastinfoset-1-2-13-sources-

jar/com/sun/xml/fastinfoset/CommonResourceBundle.java
* /opt/cola/permits/1000966784_1646171595.15/0/fastinfoset-1-2-13-sources-
jar/com/sun/xml/fastinfoset/stax/factory/StAXOutputFactory.java
* /opt/cola/permits/1000966784_1646171595.15/0/fastinfoset-1-2-13-sources-
jar/com/sun/xml/fastinfoset/tools/SAXEventSerializer.java
* /opt/cola/permits/1000966784_1646171595.15/0/fastinfoset-1-2-13-sources-
jar/com/sun/xml/fastinfoset/tools/TransformInputOutput.java
* /opt/cola/permits/1000966784_1646171595.15/0/fastinfoset-1-2-13-sources-
jar/com/sun/xml/fastinfoset/stax/events/StartElementEvent.java
* /opt/cola/permits/1000966784_1646171595.15/0/fastinfoset-1-2-13-sources-
jar/com/sun/xml/fastinfoset/dom/DOMDocumentParser.java

1.10 swagger-jersey-jaxrs 1.5.21
1.10.1 Available under license:
   Apache-2.0

1.11 json-java 20200518
1.11.1 Available under license:
============================================================================
Copyright (c) 2002 JSON.org

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

The Software shall be used for Good, not Evil.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

--------------------------------------------------------------------------
1.12 zookeeper-jute 3.5.5

1.12.1 Available under license:

Apache ZooKeeper - Jute
Copyright 2008-2019 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work.
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
    Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
    stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
    that You distribute, all copyright, patent, trademark, and
    attribution notices from the Source form of the Work,
    excluding those notices that do not pertain to any part of
    the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
your Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same “printed page” as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.13 netty-codec-mqtt 4.1.69.Final

1.13.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2021 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1235364931_1638382294.29/0/netty-codec-mqtt-4-1-69-final-sources-jar/io/netty/handler/codec/mqtt/MqttConstant.java

No license file was found, but licenses were detected in source scan.
~ Copyright 2014 The Netty Project
~
~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~
~ https://www.apache.org/licenses/LICENSE-2.0
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.

Found in path(s):
* /opt/cola/permits/1235364931_1638382294.29/0/netty-codec-mqtt-4-1-69-final-sources-jar/META-
INF/maven/io.netty/netty-codec-mqtt/pom.xml

No license file was found, but licenses were detected in source scan.

/
* Copyright 2014 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* 
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1235364931_1638382294.29/0/netty-codec-mqtt-4-1-69-final-sources-jar/io/netty/handler/codec/mqtt/MqttPubAckMessage.java
* /opt/cola/permits/1235364931_1638382294.29/0/netty-codec-mqtt-4-1-69-final-sources-jar/io/netty/handler/codec/mqtt/MqttMessageIdVariableHeader.java
* /opt/cola/permits/1235364931_1638382294.29/0/netty-codec-mqtt-4-1-69-final-sources-jar/io/netty/handler/codec/mqtt/MqttPublishMessage.java
* /opt/cola/permits/1235364931_1638382294.29/0/netty-codec-mqtt-4-1-69-final-sources-jar/io/netty/handler/codec/mqtt/MqttUnsubscribeMessage.java
* /opt/cola/permits/1235364931_1638382294.29/0/netty-codec-mqtt-4-1-69-final-sources-jar/io/netty/handler/codec/mqtt/MqttDecoder.java
* /opt/cola/permits/1235364931_1638382294.29/0/netty-codec-mqtt-4-1-69-final-sources-
/*

No license file was found, but licenses were detected in source scan.
*/
* Copyright 2014 The Netty Project
* 
* The Netty Project licenses this file to you under the Apache License, version 2.0 (the
* "License"); you may not use this file except in compliance with the License. You may obtain a
* copy of the License at:
* 
* https://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1235364931_1638382294.29/0/netty-codec-mqtt-4-1-69-final-sources-
jar/io/netty/handler/codec/mqtt/package-info.java
* /opt/cola/permits/1235364931_1638382294.29/0/netty-codec-mqtt-4-1-69-final-sources-
jar/io/netty/handler/codec/mqtt/MqttQoS.java
No license file was found, but licenses were detected in source scan.

/
* Copyright 2020 The Netty Project
* 
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* 
* https://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1235364931_1638382294.29/0/netty-codec-mqtt-4-1-69-final-sources-
jar/io/netty/handler/codec/mqtt/MqttProperties.java
* /opt/cola/permits/1235364931_1638382294.29/0/netty-codec-mqtt-4-1-69-final-sources-
jar/io/netty/handler/codec/mqtt/MqttMessageIdAndPropertiesVariableHeader.java
* /opt/cola/permits/1235364931_1638382294.29/0/netty-codec-mqtt-4-1-69-final-sources-
jar/io/netty/handler/codec/mqtt/MqttPubReplyMessageVariableHeader.java
* /opt/cola/permits/1235364931_1638382294.29/0/netty-codec-mqtt-4-1-69-final-sources-
jar/io/netty/handler/codec/mqtt/MqttSubscriptionOption.java
* /opt/cola/permits/1235364931_1638382294.29/0/netty-codec-mqtt-4-1-69-final-sources-
jar/io/netty/handler/codec/mqtt/MqttReasonCodeAndPropertiesVariableHeader.java
* /opt/cola/permits/1235364931_1638382294.29/0/netty-codec-mqtt-4-1-69-final-sources-
jar/io/netty/handler/codec/mqtt/MqttUnsubAckPayload.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2017 The Netty Project
 * 
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * 
 *   https://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1235364931_1638382294.29/0/netty-codec-mqtt-4-1-69-final-sources-
jar/io/netty/handler/codec/mqtt/MqttMessageBuilders.java

1.14 concurrentlinkedhashmap-Iru 1.3.2

1.14.1 Available under license:
Apache License, Version 2.0
FoundationProjectsPeopleGet InvolvedDownloadSupport ApacheHome » Licenses
Apache License

Version 2.0, January 2004

http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or
indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

You must give any other recipients of the Work or Derivative Works a copy of this License; and
You must cause any modified files to carry prominent notices stating that You changed the files; and
You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.
5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may...
choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

1.15 jackson-datatype-jsr310 2.13.0
1.15.1 Available under license:
This copy of Jackson JSON processor streaming parser/generator is licensed under the Apache (Software) License, version 2.0 ("the License"). See the License for details about distribution rights, and the specific rights regarding derivate works.

You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

1.16 javax-annotation-api 1.3.2
1.16.1 Available under license:
COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.0

1. Definitions.

1.1. Contributor. means each individual or entity that creates or contributes to the creation of Modifications.

1.2. Contributor Version. means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. Covered Software. means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. Executable. means the Covered Software in any form other than Source Code.

1.5. Initial Developer. means the individual or entity that first makes Original Software available under this License.

1.6. Larger Work. means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. License. means this document.

1.8. Licensable. means having the right to grant, to the maximum extent possible, whether at the time of the initial
grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. Modifications. means the Source Code and Executable form of any of the following:

   A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

   B. Any new file that contains any part of the Original Software or previous Modification; or

   C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. Original Software. means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. Patent Claims. means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code. means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. You. (or .Your.) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, .You. includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, .control. means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

   (a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

   (b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

   (c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

   (d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.
Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.
Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.
The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.
You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.
You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients' rights hereunder. You may choose to offer, and to charge a fee
for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However,
you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must
make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone,
and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the
Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.
You may distribute the Executable form of the Covered Software under the terms of this License or under the
terms of a license of Your choice, which may contain terms different from this License, provided that You are in
compliance with the terms of this License and that the license for the Executable form does not attempt to limit or
alter the recipient.s rights in the Source Code form from the rights set forth in this License. If You distribute the
Covered Software in Executable form under a different license, You must make it absolutely clear that any terms
which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby
agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or
such Contributor as a result of any such terms You offer.

You may create a Larger Work by combining Covered Software with other code not governed by the terms of
this License and distribute the Larger Work as a single product. In such a case, You must make sure the
requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.
Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License
from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no
one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.
You may always continue to use, distribute or otherwise make the Covered Software available under the terms of
the version of the License under which You originally received the Covered Software. If the Initial Developer
includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under
any subsequent version of the License, You must distribute and make the Covered Software available under the
terms of the version of the License under which You originally received the Covered Software. Otherwise, You may
also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent
version of the License published by the license steward.

4.3. Modified Versions.
When You are an Initial Developer and You want to create a new license for Your Original Software, You may
create and use a modified version of this License if You: (a) rename the license and remove any references to the
name of the license steward (except to note that the license differs from this License); and (b) otherwise make it
clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN .AS IS. BASIS, WITHOUT
WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION,
WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABILITY, FIT FOR A
PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as Participant.) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a commercial item, as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of commercial computer software. (as that term is defined at 48 C.F.R. ? 252.227-7014(a)(1)) and commercial computer software documentation. as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48
C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)

The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc. 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU
General Public License is intended to guarantee your freedom to share and change free software—to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium,
provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and
disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and
give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty
protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the
Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you
also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of
      any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the
      Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this
      License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started
      running for such interactive use in the most ordinary way, to print or display an announcement including an
      appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and
      that users may redistribute the program under these conditions, and telling the user how to view a copy of this
      License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your
      work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from
the Program, and can be reasonably considered independent and separate works in themselves, then this License,
and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute
the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on
the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and
every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather,
the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on
the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this
License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable
form under the terms of Sections 1 and 2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under
      the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more
      than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding
      source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software
c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of
that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other
system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted
interfaces, the original copyright holder who places the Program under this License may add an explicit
geographical distribution limitation excluding those countries, so that distribution is permitted only in or among
countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this
License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time
to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new
problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License
which applies to it and "any later version", you have the option of following the terms and conditions either of that
version or of any later version published by the Free Software Foundation. If the Program does not specify a version
number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are
different, write to the author to ask for permission. For software which is copyrighted by the Free Software
Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be
guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the
sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE
PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE
STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM
"AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT
NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A
PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE
PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF
ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY
COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE
PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL,
SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY
TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING
RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF
THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER
PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS
How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.

Copyright (C)

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If
this is what you want to do, use the GNU Library General Public License instead of this License.

"CLASSPATH" EXCEPTION TO THE GPL VERSION 2

Certain source files distributed by Sun Microsystems, Inc. are subject to the following clarification and special exception to the GPL Version 2, but only where Sun has expressly included in the particular source file's header the words

"Sun designates this particular file as subject to the "Classpath" exception as provided by Sun in the License file that accompanied this code."

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License Version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

1.17 hikaricp 3.4.5
1.17.1 Available under license:
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2013,2014 Brett Wooldridge
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1257259274_1642793180.98/0/hikaricp-3-4-5-sources-jar/com/zaxxer/hikari/pool/HikariPool.java
* /opt/cola/permits/1257259274_1642793180.98/0/hikaricp-3-4-5-sources-jar/com/zaxxer/hikari/HikariJNDIFactory.java
* /opt/cola/permits/1257259274_1642793180.98/0/hikaricp-3-4-5-sources-jar/com/zaxxer/hikari/metrics/dropwizard/CodaHaleMetricsTracker.java
* /opt/cola/permits/1257259274_1642793180.98/0/hikaricp-3-4-5-sources-jar/com/zaxxer/hikari/metrics/MetricsTracker.java
* /opt/cola/permits/1257259274_1642793180.98/0/hikaricp-3-4-5-sources-jar/com/zaxxer/hikari/metrics/MetricsTrackerFactory.java

No license file was found, but licenses were detected in source scan.

/
* Copyright (C) 2019 Brett Wooldridge
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1257259274_1642793180.98/0/hikaricp-3-4-5-sources-jar/com/zaxxer/hikari/util/IsolationLevel.java

No license file was found, but licenses were detected in source scan.

/
* Copyright (C) 2013 Brett Wooldridge
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2013, 2014 Brett Wooldridge
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1257259274_1642793138.98/0/hikaricp-3-4-5-sources-jar/com/zaxxer/hikari/metrics/prometheus/HikariCPCollector.java
* /opt/cola/permits/1257259274_1642793138.98/0/hikaricp-3-4-5-sources-jar/com/zaxxer/hikari/metrics/prometheus/HikariConfigMXBean.java
* /opt/cola/permits/1257259274_1642793138.98/0/hikaricp-3-4-5-sources-jar/com/zaxxer/hikari/metrics/prometheus/PrometheusHistogramMetricsTracker.java
* /opt/cola/permits/1257259274_1642793138.98/0/hikaricp-3-4-5-sources-jar/com/zaxxer/hikari/metrics/prometheus/PrometheusMetricsTracker.java
* /opt/cola/permits/1257259274_1642793138.98/0/hikaricp-3-4-5-sources-jar/com/zaxxer/hikari/metrics/dropwizard/CodahaleMetricsTrackerFactory.java
* /opt/cola/permits/1257259274_1642793138.98/0/hikaricp-3-4-5-sources-jar/com/zaxxer/hikari/util/ConcurrentBag.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2017 Brett Wooldridge
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1257259274_1642793180.98/0/hikaricp-3-4-5-sources-jar/com/zaxxer/hikari/metrics/IMetricsTracker.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2015 Brett Wooldridge
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
* http://www.apache.org/licenses/LICENSE-2.0
 *
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1257259274_1642793180.98/0/hikaricp-3-4-5-sources.jar/com/zaxxer/hikari/util/ClockSource.java
* /opt/cola/permits/1257259274_1642793180.98/0/hikaricp-3-4-5-sources.jar/com/zaxxer/hikari/metrics/PoolStats.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2014 Brett Wooldridge
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1257259274_1642793180.98/0/hikaricp-3-4-5-sources.jar/com/zaxxer/hikari/metrics/dropwizard/CodahaleHealthChecker.java
* /opt/cola/permits/1257259274_1642793180.98/0/hikaricp-3-4-5-sources.jar/com/zaxxer/hikari/pool/PoolEntry.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2013, 2014 Brett Wooldridge
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1257259274_1642793180.98/0/hikaricp-3.4-5-sources-jar/com/zaxxer/hikari/pool/PoolBase.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2016 Brett Wooldridge
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1257259274_1642793180.98/0/hikaricp-3.4-5-sources-jar/com/zaxxer/hikari/metrics/prometheus/PrometheusHistogramMetricsTrackerFactory.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2016 Brett Wooldridge
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
1.18 avro 1.10.0

1.18.1 Available under license:

Apache Avro
Copyright 2009-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.
"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
   Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
   stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
   that You distribute, all copyright, patent, trademark, and
   attribution notices from the Source form of the Work,
   excluding those notices that do not pertain to any part of
   the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.
5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.19 async-http-client-netty-utils 2.11.0
1.19.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2016 AsyncHttpClient Project. All rights reserved.
 * 
 * This program is licensed to you under the Apache License Version 2.0,
 * and you may not use this file except in compliance with the Apache License Version 2.0.
 * You may obtain a copy of the Apache License Version 2.0 at
 *     http://www.apache.org/licenses/LICENSE-2.0.
 * 
 * Unless required by applicable law or agreed to in writing,
 * software distributed under the Apache License Version 2.0 is distributed on an
 * "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the Apache License Version 2.0 for the specific language governing permissions and limitations there under.
 */

Found in path(s):
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2015 AsyncHttpClient Project. All rights reserved.
 *
 * This program is licensed to you under the Apache License Version 2.0,
 * and you may not use this file except in compliance with the Apache License Version 2.0.
 * You may obtain a copy of the Apache License Version 2.0 at
 * http://www.apache.org/licenses/LICENSE-2.0.
 *
 * Unless required by applicable law or agreed to in writing,
 * software distributed under the Apache License Version 2.0 is distributed on an
 * "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the Apache License Version 2.0 for the specific language governing permissions and limitations there under.
 */

Found in path(s):

1.20 Jackson-module-parameter-names 2.13.0
1.20.1 Available under license:
Apache-2.0

1.21 Kafka-schema-registry-client 5.3.0
1.21.1 Available under license:
No license file was found, but licenses were detected in source scan.

<project xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns="http://maven.apache.org/POM/4.0.0"
    xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/maven-v4_0_0.xsd">
    <modelVersion>4.0.0</modelVersion>

    <parent>
        <groupId>io.confluent</groupId>
        <artifactId>kafka-schema-registry-parent</artifactId>
        <version>5.3.0</version>
    </parent>

    <licenses>
        <license>
            <name>Apache License 2.0</name>
            <url>http://www.apache.org/licenses/LICENSE-2.0.html</url>
        </license>
    </licenses>
</project>
<distribution>repo</distribution>
</license>

</licenses>

<artifactId>kafka-schema-registry-client</artifactId>
<packaging>jar</packaging>
<name>kafka-schema-registry-client</name>

<dependencies>
    <dependency>
        <groupId>org.apache.kafka</groupId>
        <artifactId>kafka-clients</artifactId>
    </dependency>
    <dependency>
        <groupId>io.confluent</groupId>
        <artifactId>common-config</artifactId>
    </dependency>
    <dependency>
        <groupId>org.apache.avro</groupId>
        <artifactId>avro</artifactId>
    </dependency>
    <dependency>
        <groupId>com.fasterxml.jackson.core</groupId>
        <artifactId>jackson-databind</artifactId>
    </dependency>
    <dependency>
        <groupId>org.easymock</groupId>
        <artifactId>easymock</artifactId>
        <scope>test</scope>
    </dependency>
    <dependency>
        <groupId>org.powermock</groupId>
        <artifactId>powermock-module-junit4</artifactId>
        <scope>test</scope>
    </dependency>
    <dependency>
        <groupId>org.powermock</groupId>
        <artifactId>powermock-api-easymock</artifactId>
        <scope>test</scope>
    </dependency>
    <dependency>
        <groupId>junit</groupId>
        <artifactId>junit</artifactId>
        <scope>test</scope>
    </dependency>
</dependencies>
1.22 hibernate-validator 6.1.7.Final

1.22.1 Available under license:

Hibernate Validator, declare and validate application constraints

License: Apache License, Version 2.0
See the license.txt file in the root directory or <http://www.apache.org/licenses/LICENSE-2.0>.

Adam Stawicki
Ahmed Al Hafoudh
Alaa Nassef
Andrey Derevyanko
Andrey Rodionov
Asutosh Pandya
Benson Margulies
Brent Douglas
Carlos Vara
Carlo de Wolf
Chris Beckey
Christian Ivan
Dag Hovland
Damir Alibegovic
Dario Seidl
Davide D’Alto
Davide Marchignoli
Denis Tiago
Doug Lea
Emmanuel Bernard
Efthymis Sarbanis
Federico
Federico Mancini
Gavin King
George Gastaldi
Gerhard Petracek
Guillaume Husta
Guillaume Smet
Gunnar Morling
Hardy Ferentschik
Henno Vermeulen
Hillmer Chona
Jan-Willem Willebrands
Jason T. Greene
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION
1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work.
excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS,
## Jackson JSON processor

Jackson is a high-performance, Free/Open Source JSON processing library. It was originally written by Tatu Saloranta (tatu.salaranta@iki.fi), and has been in development since 2007. It is currently developed by a community of developers.

### Licensing

Jackson 2.x core and extension components are licensed under Apache License 2.0. To find the details that apply to this artifact see the accompanying LICENSE file.

### Credits

A list of contributors may be found from CREDITS(-2.x) file, which is included in some artifacts (usually source distributions); but is always available from the source code management (SCM) system project uses.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

    "License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

    "Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

    "Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.
"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside
You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.24 lang-tag 1.4.4

1.24.1 Available under license :
No license file was found, but licenses were detected in source scan.

/*
 * lang-tag
 * Copyright 2012-2016, Connect2id Ltd.
 */

* Licensed under the Apache License, Version 2.0 (the "License"); you may not use
This file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

*/

Found in path(s):
*/opt/cola/permits/1136256579_1613723575.96/0/lang-tag-1-4-4-sources-2-jar/com/nimbusds/langtag/LangTag.java
*/opt/cola/permits/1136256579_1613723575.96/0/lang-tag-1-4-4-sources-2-jar/com/nimbusds/langtag/ReadOnlyLangTag.java
*/opt/cola/permits/1136256579_1613723575.96/0/lang-tag-1-4-4-sources-2-jar/com/nimbusds/langtag/package-info.java
*/opt/cola/permits/1136256579_1613723575.96/0/lang-tag-1-4-4-sources-2-jar/com/nimbusds/langtag/LangTagUtils.java
*/opt/cola/permits/1136256579_1613723575.96/0/lang-tag-1-4-4-sources-2-jar/com/nimbusds/langtag/LangTagException.java

No license file was found, but licenses were detected in source scan.

<url>http://www.apache.org/licenses/LICENSE-2.0.txt</url>

Found in path(s):
*/opt/cola/permits/1136256579_1613723575.96/0/lang-tag-1-4-4-sources-2-jar/META-INF/maven/com.nimbusds/lang-tag/pom.xml

1.25 config 1.3.4

1.25.1 Available under license:

Copyright (c) 2017-2019 Truffle Blockchain Group, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.26 versly-wsdoc 1.1.010

1.27 reactive-streams 1.0.3

1.27.1 Available under license:

No license file was found, but licenses were detected in source scan.

switch to CC0

Found in path(s):
* /opt/cola/permits/1135840663_1613613063.73/0/reactive-streams-1-0-3-sources-3-jar/org/reactivestreams/Publisher.java
* /opt/cola/permits/1135840663_1613613063.73/0/reactive-streams-1-0-3-sources-3-jar/org/reactivestreams/FlowAdapters.java
* /opt/cola/permits/1135840663_1613613063.73/0/reactive-streams-1-0-3-sources-3-jar/org/reactivestreams/Processor.java
* /opt/cola/permits/1135840663_1613613063.73/0/reactive-streams-1-0-3-sources-3-jar/org/reactivestreams/Subscription.java
* /opt/cola/permits/1135840663_1613613063.73/0/reactive-streams-1-0-3-sources-3-jar/org/reactivestreams/Subscriber.java

1.28 json-path 2.4.0

1.28.1 Available under license:

No license file was found, but licenses were detected in source scan.

switch to apache

/*
 * Copyright 2011 the original author or authors.
 * Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*
Found in path(s):
* /opt/cola/permits/1199743787_1630577685.84/0/json-path-2-4-0-sources-7.jar/com/jayway/jsonpath/InvalidJsonException.java
* /opt/cola/permits/1199743787_1630577685.84/0/json-path-2-4-0-sources-7.jar/com/jayway/jsonpath/InvalidCriteriaException.java
* /opt/cola/permits/1199743787_1630577685.84/0/json-path-2-4-0-sources-7.jar/com/jayway/jsonpath/JsonPathException.java
* /opt/cola/permits/1199743787_1630577685.84/0/json-path-2-4-0-sources-7.jar/com/jayway/jsonpath/internal/path/PredicateContextImpl.java
* /opt/cola/permits/1199743787_1630577685.84/0/json-path-2-4-0-sources-7.jar/com/jayway/jsonpath/internal/Utils.java
* /opt/cola/permits/1199743787_1630577685.84/0/json-path-2-4-0-sources-7.jar/com/jayway/jsonpath/internal/function/latebinding/PathLateBindingValue.java
* /opt/cola/permits/1199743787_1630577685.84/0/json-path-2-4-0-sources-7.jar/com/jayway/jsonpath/ParseContext.java
* /opt/cola/permits/1199743787_1630577685.84/0/json-path-2-4-0-sources-7.jar/com/jayway/jsonpath/Configuration.java
* /opt/cola/permits/1199743787_1630577685.84/0/json-path-2-4-0-sources-7.jar/com/jayway/jsonpath/EvaluationListener.java
* /opt/cola/permits/1199743787_1630577685.84/0/json-path-2-4-0-sources-7.jar/com/jayway/jsonpath/internal/path/RootPathToken.java
* /opt/cola/permits/1199743787_1630577685.84/0/json-path-2-4-0-sources-7.jar/com/jayway/jsonpath/spi/json/JettisonProvider.java
* /opt/cola/permits/1199743787_1630577685.84/0/json-path-2-4-0-sources-7.jar/com/jayway/jsonpath/internal/path/CompiledPath.java
* /opt/cola/permits/1199743787_1630577685.84/0/json-path-2-4-0-sources-7.jar/com/jayway/jsonpath/WriteContext.java
* /opt/cola/permits/1199743787_1630577685.84/0/json-path-2-4-0-sources-7.jar/com/jayway/jsonpath/Option.java
* /opt/cola/permits/1199743787_1630577685.84/0/json-path-2-4-0-sources-7.jar/com/jayway/jsonpath/internal/path/PredicatePathToken.java
* /opt/cola/permits/1199743787_1630577685.84/0/json-path-2-4-0-sources-7.jar/com/jayway/jsonpath/InvalidPathException.java
* /opt/cola/permits/1199743787_1630577685.84/0/json-path-2-4-0-sources-7.jar/com/jayway/jsonpath/internal/Path.java
* /opt/cola/permits/1199743787_1630577685.84/0/json-path-2-4-0-sources-7.jar/com/jayway/jsonpath/internal/path/ArrayPathToken.java
1.29 commons-collections 3.2.1

1.29.1 Available under license:

Apache Commons Collections
Copyright 2001-2008 The Apache Software Foundation

This product includes software developed by
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.
"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.30 jersey-servlet 1.19.1

1.31 framework-tool-plugin-doc 2.1.25-SNAPSHOT
1.31.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*!
 * jQuery Color Animations v2.1.2
 * https://github.com/jquery/jquery-color
 *
 * Copyright 2013 jQuery Foundation and other contributors
 * Released under the MIT license.
 * http://jquery.org/license
 *
 * Date: Wed Jan 16 08:47:09 2013 -0600
 */

Found in path(s):

No license file was found, but licenses were detected in source scan.

/*!
 * jQuery JavaScript Library v1.7.2
 * http://jquery.com/
 *
 * Copyright 2011, John Resig
 * Dual licensed under the MIT or GPL Version 2 licenses.
 * http://jquery.org/license
 *
 * Includes Sizzle.js
 * http://sizzlejs.com/
 *
 * Copyright 2011, The Dojo Foundation
 * Released under the MIT, BSD, and GPL Licenses.
 * 
 * Date: Wed Mar 21 12:46:34 2012 -0700
 */
/*!
 * Sizzle CSS Selector Engine
 * Copyright 2011, The Dojo Foundation
 * Released under the MIT, BSD, and GPL Licenses.
 * More information: http://sizzlejs.com/
 */
// the proprietary classid attribute value (rather than the type
// proprietary methods to clear the events. Thanks to MooTools

Found in path(s):
* /opt/cola/permits/1257810994_1646348723.04/0/framework-tool-plugin-doc-2-1-25-snapshot-jar/template/html/js/jquery-1.7.2.js
1.32 dom4j 2.1.3

1.32.1 Available under license:

   No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2001-2005 (C) MetaStuff, Ltd. All Rights Reserved.
 * 
 * This software is open source.
 * See the bottom of this file for the licence.
 */

/*
 * Redistribution and use of this software and associated documentation
 * ("Software"), with or without modification, are permitted provided that the
 * following conditions are met:
 * 
 * 1. Redistributions of source code must retain copyright statements and
 *    notices. Redistributions must also contain a copy of this document.
 * 
 * 2. Redistributions in binary form must reproduce the above copyright notice,
 *    this list of conditions and the following disclaimer in the documentation
 *    and/or other materials provided with the distribution.
 * 
 * 3. The name "DOM4J" must not be used to endorse or promote products derived
 *    from this Software without prior written permission of MetaStuff, Ltd. For
 *    written permission, please contact dom4j-info@metastuff.com.
 * 
 * 4. Products derived from this Software may not be called "DOM4J" nor may
 *    "DOM4J" appear in their names without prior written permission of MetaStuff,
 *    Ltd. DOM4J is a registered trademark of MetaStuff, Ltd.
 * 
 * 5. Due credit should be given to the DOM4J Project - http://www.dom4j.org
 * 
 * THIS SOFTWARE IS PROVIDED BY METASTUFF, LTD. AND CONTRIBUTORS ``AS IS'' AND
 * ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
 * ARE DISCLAIMED. IN NO EVENT SHALL METASTUFF, LTD. OR ITS CONTRIBUTORS BE
 * LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
 * CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
 * SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
 * INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
 * CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
 * ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE
 * POSSIBILITY OF SUCH DAMAGE.
 * 
 * Copyright 2001-2005 (C) MetaStuff, Ltd. All Rights Reserved.
 */
Found in path(s):

* /opt/ws_local/PERMITS_SQL/1034543185_1588830303.8/0/dom4j-2-1-3-sources-jar/org/dom4j/tree/ElementQNameIterator.java
* /opt/ws_local/PERMITS_SQL/1034543185_1588830303.8/0/dom4j-2-1-3-sources-jar/org/dom4j/tree/ElementIterator.java
* /opt/ws_local/PERMITS_SQL/1034543185_1588830303.8/0/dom4j-2-1-3-sources-jar/org/dom4j/ElementHandler.java
* /opt/ws_local/PERMITS_SQL/1034543185_1588830303.8/0/dom4j-2-1-3-sources-jar/org/dom4j/CDATA.java
* /opt/ws_local/PERMITS_SQL/1034543185_1588830303.8/0/dom4j-2-1-3-sources-jar/org/dom4j/Document.java
* /opt/ws_local/PERMITS_SQL/1034543185_1588830303.8/0/dom4j-2-1-3-sources-jar/org/dom4j/rule/Action.java
* /opt/ws_local/PERMITS_SQL/1034543185_1588830303.8/0/dom4j-2-1-3-sources-jar/org/dom4j/rule/Pattern.java
* /opt/ws_local/PERMITS_SQL/1034543185_1588830303.8/0/dom4j-2-1-3-sources-jar/org/dom4j/rule/Rule.java
* /opt/ws_local/PERMITS_SQL/1034543185_1588830303.8/0/dom4j-2-1-3-sources-jar/org/dom4j/bean/BeanDocumentFactory.java
* /opt/ws_local/PERMITS_SQL/1034543185_1588830303.8/0/dom4j-2-1-3-sources-jar/org/dom4j/tree/FlyweightCDATA.java
* /opt/ws_local/PERMITS_SQL/1034543185_1588830303.8/0/dom4j-2-1-3-sources-jar/org/dom4j/dom/DOMDocumentType.java
* /opt/ws_local/PERMITS_SQL/1034543185_1588830303.8/0/dom4j-2-1-3-sources-jar/org/dom4j/dtd/ElementDecl.java
* /opt/ws_local/PERMITS_SQL/1034543185_1588830303.8/0/dom4j-2-1-3-sources-jar/org/dom4j/io/XMLWriter.java

Open Source Used in security_automation identity-guest-cs 225
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2001-2005 (C) MetaStuff, Ltd. All Rights Reserved.
 *
 * This software is open source.
 * See the bottom of this file for the licence.
 */

/* Redistribution and use of this software and associated documentation
 * ("Software"), with or without modification, are permitted provided that the
 * following conditions are met:
 *
 * 1. Redistributions of source code must retain copyright statements and
 * notices. Redistributions must also contain a copy of this document.
 *
 * 2. Redistributions in binary form must reproduce the above copyright notice,
 * this list of conditions and the following disclaimer in the documentation
 * and/or other materials provided with the distribution.
 *
 * 3. The name "DOM4J" must not be used to endorse or promote products derived
 * from this Software without prior written permission of MetaStuff, Ltd. For
 * written permission, please contact dom4j-info@metastuff.com.
 */
* 4. Products derived from this Software may not be called "DOM4J" nor may
* "DOM4J" appear in their names without prior written permission of MetaStuff,
* Ltd. DOM4J is a registered trademark of MetaStuff, Ltd.

* 5. Due credit should be given to the DOM4J Project - http://www.dom4j.org

* THIS SOFTWARE IS PROVIDED BY METASTUFF, LTD. AND CONTRIBUTORS ``AS IS'' AND
* ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL METASTUFF, LTD. OR ITS CONTRIBUTORS BE
* LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
* CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
* SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
* INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
* CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE
* POSSIBILITY OF SUCH DAMAGE.

* Copyright 2001-2005 (C) MetaStuff, Ltd. All Rights Reserved.

*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1034543185_1588830303.8/0/dom4j-2-1-3-sources-
jar/org/dom4j/DocumentHelper.java
* /opt/ws_local/PERMITS_SQL/1034543185_1588830303.8/0/dom4j-2-1-3-sources-
jar/org/dom4j/io/SAXReader.java
* /opt/ws_local/PERMITS_SQL/1034543185_1588830303.8/0/dom4j-2-1-3-sources-
jar/org/dom4j/io/SAXHelper.java

1.33 native-protocol 1.4.12

1.33.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.
"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "["]
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
1.3.4.1 Available under license:

Apache License

Version 2.0, January 2004

http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."
"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

You must give any other recipients of the Work or Derivative Works a copy of this License; and You must cause any modified files to carry prominent notices stating that You changed the files; and You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and
reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: HOW TO APPLY THE APACHE LICENSE TO YOUR WORK
To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
1.35 commons-text 1.9

1.35.1 Available under license:

Apache Commons Text
Copyright 2014-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (https://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).
"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate.
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.36 jnr-a64asm 1.0.0

1.36.1 Available under license :
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2018 Ossdev07
 *
 * This file is part of the JNR project.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1199743872_1630577637.85/0/jnr-a64asm-1-0-0-sources-4-jar/jnr/a64asm/Util.java
* /opt/cola/permits/1199743872_1630577637.85/0/jnr-a64asm-1-0-0-sources-4-jar/jnr/a64asm/OP.java
* /opt/cola/permits/1199743872_1630577637.85/0/jnr-a64asm-1-0-0-sources-4-jar/jnr/a64asm/Offset.java
1.37 st4 4.0.8
1.37.1 Available under license :
BSD licence
http://antlr.org/license.html

1.38 jcip-annotation 1.0.1
1.38.1 Available under license :
No license file was found, but licenses were detected in source scan.

/ *
* Copyright 2013 Stephen Connolly.
* *
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* *
* http://www.apache.org/licenses/LICENSE-2.0
* *
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1013822089_1591370825.53/0/jcip-annotations-1-0-1-sources-
  jar/net/jcip/annotations/NotThreadSafe.java
* /opt/ws_local/PERMITS_SQL/1013822089_1591370825.53/0/jcip-annotations-1-0-1-sources-
  jar/net/jcip/annotations/ThreadSafe.java
* /opt/ws_local/PERMITS_SQL/1013822089_1591370825.53/0/jcip-annotations-1-0-1-sources-
  jar/net/jcip/annotations/GuardedBy.java
* /opt/ws_local/PERMITS_SQL/1013822089_1591370825.53/0/jcip-annotations-1-0-1-sources-
  jar/net/jcip/annotations/Immutable.java

1.39 logback-core 1.2.6
1.39.1 Available under license :
Found license 'Eclipse Public License 1.0' in "* Copyright (C) 1999-2015, QOS.ch. All rights reserved. * This program and the accompanying materials are dual-licensed under * either the terms of the Eclipse Public License v1.0 as published by * under the terms of the GNU Lesser General Public License version 2.1 * as published by the
Free Software Foundation.'
Found license 'GNU Lesser General Public License' in *
Copyright (C) 1999-2015, QOS.ch. All rights reserved. *
This program and the accompanying materials are dual-licensed under *
either the terms of the Eclipse Public License v1.0 as published by *
under the terms of the GNU Lesser General Public License version 2.1 * as
published by the Free Software Foundation.'

1.40 netty-resolver 4.1.69.Final

1.40.1 Available under license:
No license was found, but licenses were detected in source scan.

/*
 * Copyright 2015 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1265673564_1643932606.25/0/netty-resolver-4-1-69-final-sources-
jar/io/netty/resolver/DefaultHostsFileEntriesResolver.java
* /opt/cola/permits/1265673564_1643932606.25/0/netty-resolver-4-1-69-final-sources-
jar/io/netty/resolver/HostsFileEntriesResolver.java
* /opt/cola/permits/1265673564_1643932606.25/0/netty-resolver-4-1-69-final-sources-
jar/io/netty/resolver/DefaultAddressResolverGroup.java
* /opt/cola/permits/1265673564_1643932606.25/0/netty-resolver-4-1-69-final-sources-
jar/io/netty/resolver/InetSocketAddressResolver.java
* /opt/cola/permits/1265673564_1643932606.25/0/netty-resolver-4-1-69-final-sources-
jar/io/netty/resolver/AbstractAddressResolver.java
* /opt/cola/permits/1265673564_1643932606.25/0/netty-resolver-4-1-69-final-sources-
jar/io/netty/resolver/AddressResolver.java
* /opt/cola/permits/1265673564_1643932606.25/0/netty-resolver-4-1-69-final-sources-
jar/io/netty/resolver/CompositeNameResolver.java
* /opt/cola/permits/1265673564_1643932606.25/0/netty-resolver-4-1-69-final-sources-
jar/io/netty/resolver/HostsFileParser.java
* /opt/cola/permits/1265673564_1643932606.25/0/netty-resolver-4-1-69-final-sources-
jar/io/netty/resolver/InetAddressResolver.java
* /opt/cola/permits/1265673564_1643932606.25/0/netty-resolver-4-1-69-final-sources-
jar/io/netty/resolver/AddressResolver.java
* /opt/cola/permits/1265673564_1643932606.25/0/netty-resolver-4-1-69-final-sources-
jar/io/netty/resolver/CompositeNameResolver.java
* /opt/cola/permits/1265673564_1643932606.25/0/netty-resolver-4-1-69-final-sources-
jar/io/netty/resolver/HostsFileParser.java
* /opt/cola/permits/1265673564_1643932606.25/0/netty-resolver-4-1-69-final-sources-
jar/io/netty/resolver/InetAddressResolver.java
* /opt/cola/permits/1265673564_1643932606.25/0/netty-resolver-4-1-69-final-sources-
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2016 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
No license file was found, but licenses were detected in source scan.

~ Copyright 2014 The Netty Project
~
~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~
~ https://www.apache.org/licenses/LICENSE-2.0
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.

Found in path(s):
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2017 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
version 2.0 (the "License"); you may not use this file except in compliance
with the License. You may obtain a copy of the License at:

https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
License for the specific language governing permissions and limitations
under the License.

Found in path(s):
* /opt/cola/permits/1265673564_1643932606.25/0/netty-resolver-4.1-69-final-sources-jar/io/netty/resolver/ResolvedAddressTypes.java
No license file was found, but licenses were detected in source scan.

*/

Copyright 2014 The Netty Project

The Netty Project licenses this file to you under the Apache License,
version 2.0 (the "License"); you may not use this file except in compliance
with the License. You may obtain a copy of the License at:

https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
License for the specific language governing permissions and limitations
under the License.

*/

Found in path(s):
* /opt/cola/permits/1265673564_1643932606.25/0/netty-resolver-4.1-69-final-sources-jar/io/netty/resolver/SimpleNameResolver.java
* /opt/cola/permits/1265673564_1643932606.25/0/netty-resolver-4.1-69-final-sources-jar/io/netty/resolver/AddressResolverGroup.java
* /opt/cola/permits/1265673564_1643932606.25/0/netty-resolver-4.1-69-final-sources-jar/io/netty/resolver/NameResolver.java
* /opt/cola/permits/1265673564_1643932606.25/0/netty-resolver-4.1-69-final-sources-jar/io/netty/resolver/NoopAddressResolverGroup.java
* /opt/cola/permits/1265673564_1643932606.25/0/netty-resolver-4.1-69-final-sources-jar/io/netty/resolver/NoopAddressResolver.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2021 The Netty Project
 * 
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * 
 * https://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1265673564_1643932606.25/0/netty-resolver-4.1.69-final-sources-
jar/io/netty/resolver/HostsFileEntriesProvider.java

1.41 failureaccess 1.0.1

1.42 httpcomponents-mime 4.5.13

1.42.1 Available under license:
Apache HttpComponents Client
Copyright 1999-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.
"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."
"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

========================================================================
This project includes Public Suffix List copied from <https://publicsuffix.org/list/effective_tld_names.dat> licensed under the terms of the Mozilla Public License, v. 2.0

Full license text: <http://mozilla.org/MPL/2.0/>

Mozilla Public License Version 2.0
==================================

1. Definitions
---------------

1.1. "Contributor"
means each individual or legal entity that creates, contributes to the creation of, or owns Covered Software.

1.2. "Contributor Version"
means the combination of the Contributions of others (if any) used by a Contributor and that particular Contributor's Contribution.

1.3. "Contribution"
means Covered Software of a particular Contributor.

1.4. "Covered Software"
means Source Code Form to which the initial Contributor has attached the notice in Exhibit A, the Executable Form of such Source Code
Form, and Modifications of such Source Code Form, in each case including portions thereof.

1.5. "Incompatible With Secondary Licenses"
means

(a) that the initial Contributor has attached the notice described in Exhibit B to the Covered Software; or

(b) that the Covered Software was made available under the terms of version 1.1 or earlier of the License, but not also under the terms of a Secondary License.

1.6. "Executable Form"
means any form of the work other than Source Code Form.

1.7. "Larger Work"
means a work that combines Covered Software with other material, in a separate file or files, that is not Covered Software.

1.8. "License"
means this document.

1.9. "Licensable"
means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently, any and all of the rights conveyed by this License.

1.10. "Modifications"
means any of the following:

(a) any file in Source Code Form that results from an addition to, deletion from, or modification of the contents of Covered Software; or

(b) any new file in Source Code Form that contains any Covered Software.

1.11. "Patent Claims" of a Contributor
means any patent claim(s), including without limitation, method, process, and apparatus claims, in any patent Licensable by such Contributor that would be infringed, but for the grant of the License, by the making, using, selling, offering for sale, having made, import, or transfer of either its Contributions or its Contributor Version.

1.12. "Secondary License"
means either the GNU General Public License, Version 2.0, the GNU
Lesser General Public License, Version 2.1, the GNU Affero General Public License, Version 3.0, or any later versions of those licenses.

1.13. "Source Code Form"
means the form of the work preferred for making modifications.

1.14. "You" (or "Your")
means an individual or a legal entity exercising rights under this License. For legal entities, "You" includes any entity that controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants and Conditions

2.1. Grants

Each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by such Contributor to use, reproduce, make available, modify, display, perform, distribute, and otherwise exploit its Contributions, either on an unmodified basis, with Modifications, or as part of a Larger Work; and

(b) under Patent Claims of such Contributor to make, use, sell, offer for sale, have made, import, and otherwise transfer either its Contributions or its Contributor Version.

2.2. Effective Date

The licenses granted in Section 2.1 with respect to any Contribution become effective for each Contribution on the date the Contributor first distributes such Contribution.

2.3. Limitations on Grant Scope

The licenses granted in this Section 2 are the only rights granted under this License. No additional rights or licenses will be implied from the distribution or licensing of Covered Software under this License. Notwithstanding Section 2.1(b) above, no patent license is granted by a Contributor:
(a) for any code that a Contributor has removed from Covered Software; or

(b) for infringements caused by: (i) Your and any other third party's modifications of Covered Software, or (ii) the combination of its Contributions with other software (except as part of its Contributor Version); or

(c) under Patent Claims infringed by Covered Software in the absence of its Contributions.

This License does not grant any rights in the trademarks, service marks, or logos of any Contributor (except as may be necessary to comply with the notice requirements in Section 3.4).

2.4. Subsequent Licenses

No Contributor makes additional grants as a result of Your choice to distribute the Covered Software under a subsequent version of this License (see Section 10.2) or under the terms of a Secondary License (if permitted under the terms of Section 3.3).

2.5. Representation

Each Contributor represents that the Contributor believes its Contributions are its original creation(s) or it has sufficient rights to grant the rights to its Contributions conveyed by this License.

2.6. Fair Use

This License is not intended to limit any rights You have under applicable copyright doctrines of fair use, fair dealing, or other equivalents.

2.7. Conditions

Sections 3.1, 3.2, 3.3, and 3.4 are conditions of the licenses granted in Section 2.1.

3. Responsibilities

3.1. Distribution of Source Form

All distribution of Covered Software in Source Code Form, including any Modifications that You create or to which You contribute, must be under the terms of this License. You must inform recipients that the Source
Code Form of the Covered Software is governed by the terms of this License, and how they can obtain a copy of this License. You may not attempt to alter or restrict the recipients' rights in the Source Code Form.

3.2. Distribution of Executable Form

If You distribute Covered Software in Executable Form then:

(a) such Covered Software must also be made available in Source Code Form, as described in Section 3.1, and You must inform recipients of the Executable Form how they can obtain a copy of such Source Code Form by reasonable means in a timely manner, at a charge no more than the cost of distribution to the recipient; and

(b) You may distribute such Executable Form under the terms of this License, or sublicense it under different terms, provided that the license for the Executable Form does not attempt to limit or alter the recipients' rights in the Source Code Form under this License.

3.3. Distribution of a Larger Work

You may create and distribute a Larger Work under terms of Your choice, provided that You also comply with the requirements of this License for the Covered Software. If the Larger Work is a combination of Covered Software with a work governed by one or more Secondary Licenses, and the Covered Software is not Incompatible With Secondary Licenses, this License permits You to additionally distribute such Covered Software under the terms of such Secondary License(s), so that the recipient of the Larger Work may, at their option, further distribute the Covered Software under the terms of either this License or such Secondary License(s).

3.4. Notices

You may not remove or alter the substance of any license notices (including copyright notices, patent notices, disclaimers of warranty, or limitations of liability) contained within the Source Code Form of the Covered Software, except that You may alter any license notices to the extent required to remedy known factual inaccuracies.

3.5. Application of Additional Terms

You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, You may do so only on Your own behalf, and not on behalf of any Contributor. You must make it absolutely clear that any such warranty, support, indemnity, or liability obligation is offered by
You alone, and You hereby agree to indemnify every Contributor for any liability incurred by such Contributor as a result of warranty, support, indemnity or liability terms You offer. You may include additional disclaimers of warranty and limitations of liability specific to any jurisdiction.

4. Inability to Comply Due to Statute or Regulation

If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Software due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be placed in a text file included with all distributions of the Covered Software under this License. Except to the extent prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to understand it.

5. Termination

5.1. The rights granted under this License will terminate automatically if You fail to comply with any of its terms. However, if You become compliant, then the rights granted under this License from a particular Contributor are reinstated (a) provisionally, unless and until such Contributor explicitly and finally terminates Your grants, and (b) on an ongoing basis, if such Contributor fails to notify You of the non-compliance by some reasonable means prior to 60 days after You have come back into compliance. Moreover, Your grants from a particular Contributor are reinstated on an ongoing basis if such Contributor notifies You of the non-compliance by some reasonable means, this is the first time You have received notice of non-compliance with this License from such Contributor, and You become compliant prior to 30 days after Your receipt of the notice.

5.2. If You initiate litigation against any entity by asserting a patent infringement claim (excluding declaratory judgment actions, counter-claims, and cross-claims) alleging that a Contributor Version directly or indirectly infringes any patent, then the rights granted to You by any and all Contributors for the Covered Software under Section 2.1 of this License shall terminate.

5.3. In the event of termination under Sections 5.1 or 5.2 above, all end user license agreements (excluding distributors and resellers) which have been validly granted by You or Your distributors under this License prior to termination shall survive termination.
6. Disclaimer of Warranty

Covered Software is provided under this License on an "as is" basis, without warranty of any kind, either expressed, implied, or statutory, including, without limitation, warranties that the Covered Software is free of defects, merchantable, fit for a particular purpose or non-infringing. The entire risk as to the quality and performance of the Covered Software is with You. Should any Covered Software prove defective in any respect, You (not any Contributor) assume the cost of any necessary servicing, repair, or correction. This disclaimer of warranty constitutes an essential part of this License. No use of any Covered Software is authorized under this License except under this disclaimer.

7. Limitation of Liability

Under no circumstances and under no legal theory, whether tort (including negligence), contract, or otherwise, shall any Contributor, or anyone who distributes Covered Software as permitted above, be liable to You for any direct, indirect, special, incidental, or consequential damages of any character including, without limitation, damages for lost profits, loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses, even if such party shall have been informed of the possibility of such damages. This limitation of liability shall not apply to liability for death or personal injury resulting from such party's negligence to the extent applicable law prohibits such limitation. Some jurisdictions do not allow the exclusion or limitation of incidental or consequential damages, so this exclusion and limitation may not apply to You.

8. Litigation

Any litigation relating to this License may be brought only in the courts of a jurisdiction where the defendant maintains its principal place of business and such litigation shall be governed by laws of that
jurisdiction, without reference to its conflict-of-law provisions. Nothing in this Section shall prevent a party's ability to bring cross-claims or counter-claims.

9. Miscellaneous

This License represents the complete agreement concerning the subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not be used to construe this License against a Contributor.

10. Versions of the License

Mozilla Foundation is the license steward. Except as provided in Section 10.3, no one other than the license steward has the right to modify or publish new versions of this License. Each version will be given a distinguishing version number.

10.2. Effect of New Versions

You may distribute the Covered Software under the terms of the version of the License under which You originally received the Covered Software, or under the terms of any subsequent version published by the license steward.

10.3. Modified Versions

If you create software not governed by this License, and you want to create a new license for such software, you may create and use a modified version of this License if you rename the license and remove any references to the name of the license steward (except to note that such modified license differs from this License).

10.4. Distributing Source Code Form that is Incompatible With Secondary Licenses

If You choose to distribute Source Code Form that is Incompatible With Secondary Licenses under the terms of this version of the License, the notice described in Exhibit B of this License must be attached.

Exhibit A - Source Code Form License Notice
This Source Code Form is subject to the terms of the Mozilla Public License, v. 2.0. If a copy of the MPL was not distributed with this file, You can obtain one at http://mozilla.org/MPL/2.0/.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

Exhibit B - "Incompatible With Secondary Licenses" Notice

---------------------------------------------------------

This Source Code Form is "Incompatible With Secondary Licenses", as defined by the Mozilla Public License, v. 2.0.

1.43 jnr-ffi 2.2.0

1.43.1 Available under license:
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2013 Wayne Meissner
 *
 * This file is part of the JNR project.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *    http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
  * /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/provider/ToNativeType.java
  * /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/provider/ResultType.java
  * /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/provider/jffi/Types.java
  * /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/provider/NativeVariable.java
  * /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-
jar/jnr/ffi/mapper/SimpleTypeMapper.java
*/
*/opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-/*
jar/jnr/ffi/provider/jffi/ConverterMetaData.java
*/
*/opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-/*
jar/jnr/ffi/provider/SigType.java
*/
*/opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-/*
jar/jnr/ffi/mapper/SimpleFunctionMapper.java
*/
*/opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-/*
jar/jnr/ffi/provider/jffi/StructByReferenceResultConverterFactory.java
*/
*/opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-/*
jar/jnr/ffi/provider/jffi/AnnotationTypeMapper.java
*/
*/opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-/*
jar/jnr/ffi/provider/FromNativeType.java
*/
*/opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-/*
jar/jnr/ffi/provider/ParameterType.java
*/
*/opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-/*
jar/jnr/ffi/provider/InterfaceScanner.java
*/
*/opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-/*
jar/jnr/ffi/provider/NativeFunction.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2008-2011 Wayne Meissner
 *
 * This file is part of the JNR project.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
*/
*/opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-/*
jar/jnr/ffi/provider/FFIProvider.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2008-2010 Wayne Meissner
 *
 * This file is part of the JNR project.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
/*
* Some of the design and code of this class is from the javolution project.
*
* Copyright (C) 2006 - Javolution (http://javolution.org/)
* All rights reserved.
*
* Permission to use, copy, modify, and distribute this software is
* freely granted, provided that this notice is preserved.
*/

Found in path(s):
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/Struct.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/Union.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2016 Wayne Meissner
*
* This file is part of the JNR project.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* *
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-
  jar/jnr/ffi/provider/ffi/platform/aarch64/linux/TypeAliases.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2012 Wayne Meissner
*

* This file is part of the JNR project.
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/types/int32_t.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/provider/converters/BoxedFloatArrayParameterConverter.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/provider/jffi/NativeLibraryLoader.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/provider/jffi/InvokerTypeMapper.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/provider/jffi/PrimitiveArrayParameterStrategy.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/provider/jffi/LocalVariable.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/provider/jffi/ReflectionVariableAccessorGenerator.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/provider/jffi/LibraryLoader.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/provider/jffi/platform/x86_64/solaris/TypeAliases.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/provider/jffi/platform/s390x/linux/TypeAliases.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/provider/jffi/platform/ppc/darwin/TypeAliases.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/provider/jffi/platform/ppc64/aix/TypeAliases.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/provider/jffi/platform/x86_64/openbsd/TypeAliases.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/provider/jffi/InvalidProvider.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/provider/jffi/ReflectionVariableAccessorGenerator.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/provider/jffi/LibraryLoader.java
jar/jffi/provider/jffi/platform/TypeAliases.java
* /opt/cola/permits/1199743715_1630577646.23/0/jffi/provider/jffi/mapper/FromNativeTypes.java
* /opt/cola/permits/1199743715_1630577646.23/0/jffi/provider/jffi/NoTrace.java
* /opt/cola/permits/1199743715_1630577646.23/0/jffi/provider/jffi/NativeFinalizer.java
* /opt/cola/permits/1199743715_1630577646.23/0/jffi/provider/converters/Pointer64ArrayParameterConverter.java
* /opt/cola/permits/1199743715_1630577646.23/0/jffi/provider/converters/StringBufferParameterConverter.java
* /opt/cola/permits/1199743715_1630577646.23/0/jffi/provider/converters/BoxedLong64ArrayParameterConverter.java
* /opt/cola/permits/1199743715_1630577646.23/0/jffi/provider/jffi/platform/ppc/linux/TypeAliases.java
* /opt/cola/permits/1199743715_1630577646.23/0/jffi/provider/jffi/types/gid_t.java
* /opt/cola/permits/1199743715_1630577646.23/0/jffi/provider/jffi/SimpleNativeContext.java
* /opt/cola/permits/1199743715_1630577646.23/0/jffi/provider/jffi/BufferParameterStrategy.java
* /opt/cola/permits/1199743715_1630577646.23/0/jffi/provider/jffi/types/in_addr_t.java
* /opt/cola/permits/1199743715_1630577646.23/0/jffi/provider/jffi/types/key_t.java
* /opt/cola/permits/1199743715_1630577646.23/0/jffi/provider/jffi/types/uid_t.java
* /opt/cola/permits/1199743715_1630577646.23/0/jffi/provider/jffi/VariableAccessorGenerator.java
/*
* Copyright (C) 2017 Oracle and/or its affiliates.
* 
* This file is part of the JNR project.
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/provider/jffi/platform/arm/linux/TypeAliases.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2008-2012 Wayne Meissner
* 
* This file is part of the JNR project.
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/provider/jffi/platform/sparcv9/linux/TypeAliases.java
No license file was found, but licenses were detected in source scan.
/*
 * Copyright (C) 2007 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/util/ref/FinalizableWeakReference.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/util/ref/FinalizablePhantomReference.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/util/ref/FinalizableReferenceQueue.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/util/ref/FinalizableReference.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/util/ref/FinalizableSoftReference.java

/*
 * Copyright (C) 2010 Wayne Meissner
 *
 * This file is part of the JNR project.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/byref/AbstractReference.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/Memory.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/byref/AbstractNumberReference.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/Runtime.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/Type.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2010 The Miyamoto Team
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/util/AnnotationProperty.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/util/AnnotationProxy.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2008-2010 Wayne Meissner
*
* This file is part of the JNR project.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/provider/jffi/ArrayMemoryIO.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/provider/jffi/InvokerUtil.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/annotations/Out.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/provider/jffi/AbstractX86StubCompiler.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/provider/jffi/LibraryLoader.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/provider/AbstractRuntime.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/provider/jffi/StubCompiler.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/LastError.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/provider/jffi/NativeRuntime.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/annotations/SaveError.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/mapper/TypeMapper.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/provider/jffi/JNINativeInterface.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/provider/jffi/AsmClassLoader.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/annotations/Pinned.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/annotations/In.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/Address.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/provider/ShareMemoryIO.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/provider/jffi/Provider.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/provider/jffi/AbstractA64StubCompiler.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/annotations/Direct.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/provider/jffi/NumberUtil.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/provider/jffi/mapper/MethodResultContext.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/provider/jffi/ByReference.java
Open Source Used in security_automation identity-guest-cs 274
jar/jnr/ffi/provider/jffi/SymbolNotFoundError.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/util/EnumMapper.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/byref/ByteByReference.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/annotations/Clear.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/provider/InAccessibleMemoryIO.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/provider/NullMemoryIO.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/LibraryOption.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/provider/NullTypeMapper.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/byref/FloatByReference.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/provider/AbstractMemoryIO.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/mapper/MethodParameterContext.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/annotations/StdCall.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/provider/Invoker.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/byref/IntByReference.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/provider/LoadedLibrary.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/provider/NullLibrary.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/Library.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/provider/AbstractMemoryIO.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/provider/InAccessibleMemoryIO.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2011 Wayne Meissner
 *
 * This file is part of the JNR project.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2009-2010 Wayne Meissner
 *
 * This file is part of the JNR project.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *    http://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/NativeType.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2014 Wayne Meissner
* *
* This file is part of the JNR project.
* *
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* *
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/provider/jffi/platform/ppc64/linux/TypeAliases.java
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/provider/jffi/platform/ppc64le/linux/TypeAliases.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2008 The Guava Authors
* *
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* *
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. 
* See the License for the specific language governing permissions and 
* limitations under the License. 
*/

Found in path(s):
* /opt/cola/permits/1199743715_1630577646.23/0/jnr-ffi-2-2-0-sources-2-jar/jnr/ffi/util/ref/internal/Finalizer.java

1.44 netty-codec 4.1.69.Final

1.44.1 Available under license :
No license file was found, but licenses were detected in source scan.

*/
*
* Copyright 2012 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License, 
* version 2.0 (the "License"); you may not use this file except in compliance 
* with the License. You may obtain a copy of the License at:
* 
*   https://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software 
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT 
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the 
* License for the specific language governing permissions and limitations 
* under the License.
*/
*/
*
* Written by Robert Harder and released to the public domain, as explained at 
* https://creativecommons.org/licenses/publicdomain
*/
/**
* Utility class for \{@link ByteBuf\} that encodes and decodes to and from 
* \<a href="https://en.wikipedia.org/wiki/Base64">Base64</a> notation. 
* \</p>
* The encoding and decoding algorithm in this class has been derived from 
* \<a href="http://iharder.sourceforge.net/current/java/base64/">Robert Harder's Public Domain 
* Base64 Encoder/Decoder</a>. 
*/

Found in path(s):
* /opt/cola/permits/1265673536_1643922096.16/0/netty-codec-4-1-69-final-sources- 
* jar/io/netty/handler/codec/base64/Base64.java
No license file was found, but licenses were detected in source scan.

/
* Copyright 2016 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1265673536_1643922096.16/0/netty-codec-4-1-69-final-sources-
  jar/io/netty/handler/codec/compression/ByteBufChecksum.java
* /opt/cola/permits/1265673536_1643922096.16/0/netty-codec-4-1-69-final-sources-
  jar/io/netty/handler/codec/compression/CompressionUtil.java
* /opt/cola/permits/1265673536_1643922096.16/0/netty-codec-4-1-69-final-sources-
  jar/io/netty/handler/codec/DatagramPacketDecoder.java
* /opt/cola/permits/1265673536_1643922096.16/0/netty-codec-4-1-69-final-sources-
  jar/io/netty/handler/codec/string/LineEncoder.java
* /opt/cola/permits/1265673536_1643922096.16/0/netty-codec-4-1-69-final-sources-
  jar/io/netty/handler/codec/CodecOutputList.java
* /opt/cola/permits/1265673536_1643922096.16/0/netty-codec-4-1-69-final-sources-
  jar/io/netty/handler/codec/DateFormatter.java
* /opt/cola/permits/1265673536_1643922096.16/0/netty-codec-4-1-69-final-sources-
  jar/io/netty/handler/codec/string/LineSeparator.java
* /opt/cola/permits/1265673536_1643922096.16/0/netty-codec-4-1-69-final-sources-
  jar/io/netty/handler/codec/DatagramPacketEncoder.java

No license file was found, but licenses were detected in source scan.

~ Copyright 2012 The Netty Project
~
~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~
~ https://www.apache.org/licenses/LICENSE

~
~ 2.0
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.
/*
 * Copyright 2012 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

/*
 * Found in path(s):
 * /opt/cola/permits/1265673536_1643922096.16/0/netty-codec-4-1-69-final-sources-jar/io/netty/handler/codec/compression/CompressionException.java
 * /opt/cola/permits/1265673536_1643922096.16/0/netty-codec-4-1-69-final-sources-jar/io/netty/handler/codec/compression/LengthFieldPrepender.java
 * /opt/cola/permits/1265673536_1643922096.16/0/netty-codec-4-1-69-final-sources-jar/io/netty/handler/codec/marshalling/ThreadLocalUnmarshallerProvider.java
 * /opt/cola/permits/1265673536_1643922096.16/0/netty-codec-4-1-69-final-sources-jar/io/netty/handler/codec/marshalling/MarshallingEncoder.java
 * /opt/cola/permits/1265673536_1643922096.16/0/netty-codec-4-1-69-final-sources-jar/io/netty/handler/codec/marshalling/MarshallerProvider.java
 * /opt/cola/permits/1265673536_1643922096.16/0/netty-codec-4-1-69-final-sources-jar/io/netty/handler/codec/serialization/CompatibleObjectEncoder.java
 * /opt/cola/permits/1265673536_1643922096.16/0/netty-codec-4-1-69-final-sources-jar/io/netty/handler/codec/ByteToMessageCodec.java
 * /opt/cola/permits/1265673536_1643922096.16/0/netty-codec-4-1-69-final-sources-jar/io/netty/handler/codec/string/StringDecoder.java
 * /opt/cola/permits/1265673536_1643922096.16/0/netty-codec-4-1-69-final-sources-jar/io/netty/handler/codec/marshalling/MarshallerProvider.java
 * /opt/cola/permits/1265673536_1643922096.16/0/netty-codec-4-1-69-final-sources-jar/io/netty/handler/codec/TooLongFrameException.java
 * /opt/cola/permits/1265673536_1643922096.16/0/netty-codec-4-1-69-final-sources-jar/io/netty/handler/codec/marshalling/DefaultMarshallerProvider.java
 * /opt/cola/permits/1265673536_1643922096.16/0/netty-codec-4-1-69-final-sources-jar/io/netty/handler/codec/ByteToMessageCodec.java
 * /opt/cola/permits/1265673536_1643922096.16/0/netty-codec-4-1-69-final-sources-jar/io/netty/handler/codec/marshalling/DefaultUnmarshallerProvider.java
 */

No license file was found, but licenses were detected in source scan.
Copyright 2015 The Netty Project

The Netty Project licenses this file to you under the Apache License, version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at:

https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either...
express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1265673536_1643922096.16/0/netty-codec-4-1-69-final-sources-
  jar/io/netty/handler/codec/ValueConverter.java
* /opt/cola/permits/1265673536_1643922096.16/0/netty-codec-4-1-69-final-sources-
  jar/io/netty/handler/codec/DefaultHeadersImpl.java
* /opt/cola/permits/1265673536_1643922096.16/0/netty-codec-4-1-69-final-sources-
  jar/io/netty/handler/codec/CharSequenceValueConverter.java
No license file was found, but licenses were detected in source scan.

/ *
* Copyright 2012 The Netty Project
* 
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* 
*   https://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

/*
* Written by Robert Harder and released to the public domain, as explained at
* https://creativecommons.org/licenses/publicdomain
*/

/*
* Enumeration of supported Base64 dialects.
* <p>
* The internal lookup tables in this class has been derived from
* <a href="http://iharder.sourceforge.net/current/java/base64/"">Robert Harder's Public Domain
* Base64 Encoder/Decoder</a>.
*/

Found in path(s):
* /opt/cola/permits/1265673536_1643922096.16/0/netty-codec-4-1-69-final-sources-
  jar/io/netty/handler/codec/base64/Base64Dialect.java
No license file was found, but licenses were detected in source scan.

/ *
* Copyright 2012 The Netty Project
The Netty Project licenses this file to you under the Apache License, version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at:

https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

A decoder that splits the received ByteBufs dynamically by the value of the length field in the message. It is particularly useful when you decode a binary message which has an integer header field that represents the length of the message body or the whole message.

@@LengthFieldBasedFrameDecoder@@ has many configuration parameters so that it can decode any message with a length field, which is often seen in proprietary client-server protocols. Here are some example that will give you the basic idea on which option does what.

**<h3>2 bytes length field at offset 0, do not strip header</h3>**

The value of the length field in this example is 12 (0x0C) which represents the length of "HELLO, WORLD". By default, the decoder assumes that the length field represents the number of the bytes that follows the length field. Therefore, it can be decoded with the simplistic parameter combination.

```pre```
<table>
<thead>
<tr>
<th>Length</th>
<th>Actual Content</th>
</tr>
</thead>
<tbody>
<tr>
<td>0x000C</td>
<td>&quot;HELLO, WORLD&quot;</td>
</tr>
</tbody>
</table>
```

**<h3>2 bytes length field at offset 0, strip header</h3>**

Because we can get the length of the content by calling ByteBuf#readableBytes(), you might want to strip the length.
field by specifying `<tt>initialBytesToStrip</tt>`. In this example, we specified `<tt>2</tt>`, that is same with the length of the length field, to strip the first two bytes.

```
* lengthFieldOffset   = 0
* lengthFieldLength   = 2
* lengthAdjustment    = 0
* `<b>initialBytesToStrip</b>` = `<b>2</b>` (= the length of the Length field)
```

* BEFORE DECODE (14 bytes)   AFTER DECODE (12 bytes)
```
* +--------+----------------+      +----------------+
* | Length | Actual Content |----->| Actual Content |
* | 0x000C | "HELLO, WORLD" |      | "HELLO, WORLD" |
* +--------+----------------+      +----------------+
```

* <h3>2 bytes length field at offset 0, do not strip header, the length field represents the length of the whole message</h3>

* In most cases, the length field represents the length of the message body only, as shown in the previous examples. However, in some protocols, the length field represents the length of the whole message, including the message header. In such a case, we specify a non-zero `<tt>lengthAdjustment</tt>`. Because the length value in this example message is always greater than the body length by `<tt>2</tt>`, we specify `<tt>-2</tt>` as `<tt>lengthAdjustment</tt>` for compensation.

```
* lengthFieldOffset   = 0
* lengthFieldLength   = 2
* `<b>lengthAdjustment</b>` = `<b>-2</b>` (= the length of the Length field)
* initialBytesToStrip = 0
```

* BEFORE DECODE (14 bytes)   AFTER DECODE (14 bytes)
```
* +--------+----------------+      +--------+----------------+
* | Length | Actual Content |----->| Length | Actual Content |
* | 0x000E | "HELLO, WORLD" |      | 0x000E | "HELLO, WORLD" |
* +--------+----------------+      +--------+----------------+
```

* <h3>3 bytes length field at the end of 5 bytes header, do not strip header</h3>

* The following message is a simple variation of the first example. An extra header value is prepended to the message. `<tt>lengthAdjustment</tt>` is zero again because the decoder always takes the length of the prepended data into account during frame length calculation.

```
* `<b>lengthFieldOffset</b>` = `<b>2</b>` (= the length of Header 1)
* `<b>lengthFieldLength</b>` = `<b>3</b>`
* lengthAdjustment = 0
* initialBytesToStrip = 0
* 
* BEFORE DECODE (17 bytes) AFTER DECODE (17 bytes)
* +---------------------------------+---------------------------------+
* | Header 1 | Length | Actual Content |----->| Header 1 | Length | Actual Content |
* | 0xCAFE | 0x00000C | "HELLO, WORLD" |      | 0xCAFE | 0x00000C | "HELLO, WORLD" |
* +---------------------------------+---------------------------------+
* </pre>
* 
* <h3>3 bytes length field at the beginning of 5 bytes header, do not strip header</h3>
* 
* This is an advanced example that shows the case where there is an extra header between the length field and the message body. You have to specify a positive <tt>lengthAdjustment</tt> so that the decoder counts the extra header into the frame length calculation.

* undefined
* 
* lengthFieldOffset = 0
* lengthFieldLength = 3
* <b>lengthAdjustment</b> = <b>2</b> (= the length of Header 1)
* initialBytesToStrip = 0
* 
* BEFORE DECODE (17 bytes) AFTER DECODE (17 bytes)
* +---------------------------------+---------------------------------+
* | Length | Header 1 | Actual Content |----->| Length | Header 1 | Actual Content |
* | 0x00000C | 0xCAFE | "HELLO, WORLD" |      | 0x00000C | 0xCAFE | "HELLO, WORLD" |
* +---------------------------------+---------------------------------+
* </pre>
* 
* <h3>2 bytes length field at offset 1 in the middle of 4 bytes header, strip the first header field and the length field</h3>
* 
* This is a combination of all the examples above. There are the prepended header before the length field and the extra header after the length field. The prepended header affects the <tt>lengthFieldOffset</tt> and the extra header affects the <tt>lengthAdjustment</tt>. We also specified a non-zero <tt>initialBytesToStrip</tt> to strip the length field and the prepended header from the frame. If you don't want to strip the prepended header, you could specify <tt>0</tt> for <tt>initialBytesToStrip</tt>.

* undefined
* 
* lengthFieldOffset = 1 (= the length of HDR1)
* lengthFieldLength = 2
* <b>lengthAdjustment</b> = <b>1</b> (= the length of HDR2)
* <b>initialBytesToStrip</b> = <b>3</b> (= the length of HDR1 + LEN)
* 
* BEFORE DECODE (16 bytes) AFTER DECODE (13 bytes)
* +---------------------------------+---------------------------------+
* | HDR1 | Length | HDR2 | Actual Content |----->| HDR2 | Actual Content |
* +---------------------------------+---------------------------------+
* | 0xCA | 0x000C | 0xFE | "HELLO, WORLD" | | 0xFE | "HELLO, WORLD" |
* +------------------------------------+------------------------------------+
* </pre>

* ch3>2 bytes length field at offset 1 in the middle of 4 bytes header,
* strip the first header field and the length field, the length field
* represents the length of the whole message</h3>
*
* Let's give another twist to the previous example. The only difference from
* the previous example is that the length field represents the length of the
* whole message instead of the message body, just like the third example.
* We have to count the length of HDR1 and Length into '<tt>lengthAdjustment</tt>'.
* Please note that we don't need to take the length of HDR2 into account
* because the length field already includes the whole header length.
* <pre>
* lengthFieldOffset = 1
* lengthFieldLength = 2
* <b>lengthAdjustment</b> = <b>-3</b> (= the length of HDR1 + LEN, negative)
* <b>initialBytesToStrip</b> = <b>3</b>
*
* BEFORE DECODE (16 bytes)                        AFTER DECODE (13 bytes)
* +------------------------------------+------------------------------------+
* | HDR1 | Length | HDR2 | Actual Content |----->| HDR2 | Actual Content |
* | 0xCA | 0x0010 | 0xFE | "HELLO, WORLD" |      | 0xFE | "HELLO, WORLD" |
* +------------------------------------+------------------------------------+
* </pre>
* @see LengthFieldPrepender
* /

Found in path(s):
* /opt/cola/permits/1265673536_1643922096.16/0/netty-codec-4.1.69-final-sources-jar/io/netty/handler/codec/LengthFieldBasedFrameDecoder.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2019 The Netty Project
* 
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* 
* https://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/
Found in path(s):
* /opt/cola/permits/1265673536_1643922096.16/0/netty-codec-4-1-69-final-sources-jar/io/netty/handler/codec/compression/Lz4XXHash32.java
No license file was found, but licenses were detected in source scan.

*/

* Copyright 2014 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License, version 2.0 (the
* "License"); you may not use this file except in compliance with the License. You may obtain a
* copy of the License at:
*
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1265673536_1643922096.16/0/netty-codec-4-1-69-final-sources-jar/io/netty/handler/codec/EmptyHeaders.java
* /opt/cola/permits/1265673536_1643922096.16/0/netty-codec-4-1-69-final-sources-jar/io/netty/handler/codec/Headers.java
* /opt/cola/permits/1265673536_1643922096.16/0/netty-codec-4-1-69-final-sources-jar/io/netty/handler/codec/DefaultHeaders.java
No license file was found, but licenses were detected in source scan.

*/

* Copyright 2021 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/
Found in path(s):
  */opt/cola/permits/1265673536_1643922096.16/0/netty-codec-4-1-69-final-sources-
  jar/io/netty/handler/codec/compression/ZstdConstants.java
  */opt/cola/permits/1265673536_1643922096.16/0/netty-codec-4-1-69-final-sources-
  jar/io/netty/handler/codec/compression/ZstdEncoder.java
  */opt/cola/permits/1265673536_1643922096.16/0/netty-codec-4-1-69-final-sources-
  jar/io/netty/handler/codec/compression/Brotli.java
  */opt/cola/permits/1265673536_1643922096.16/0/netty-codec-4-1-69-final-sources-
  jar/io/netty/handler/codec/compression/Zstd.java
  */opt/cola/permits/1265673536_1643922096.16/0/netty-codec-4-1-69-final-sources-
  jar/io/netty/handler/codec/compression/BrotliEncoder.java
  */opt/cola/permits/1265673536_1643922096.16/0/netty-codec-4-1-69-final-sources-
  jar/io/netty/handler/codec/compression/BrotliOptions.java
  */opt/cola/permits/1265673536_1643922096.16/0/netty-codec-4-1-69-final-sources-
  jar/io/netty/handler/codec/compression/DeflateOptions.java
  */opt/cola/permits/1265673536_1643922096.16/0/netty-codec-4-1-69-final-sources-
  jar/io/netty/handler/codec/compression/StandardCompressionOptions.java
  */opt/cola/permits/1265673536_1643922096.16/0/netty-codec-4-1-69-final-sources-
  jar/io/netty/handler/codec/compression/BrotliDecoder.java
  */opt/cola/permits/1265673536_1643922096.16/0/netty-codec-4-1-69-final-sources-
  jar/io/netty/handler/codec/compression/CompressionOptions.java
  */opt/cola/permits/1265673536_1643922096.16/0/netty-codec-4-1-69-final-sources-
  jar/io/netty/handler/codec/compression/GzipOptions.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2013 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
  */opt/cola/permits/1265673536_1643922096.16/0/netty-codec-4-1-69-final-sources-
  jar/io/netty/handler/codec/xml/XmlFrameDecoder.java
  */opt/cola/permits/1265673536_1643922096.16/0/netty-codec-4-1-69-final-sources-
  jar/io/netty/handler/codec/compression/Crc32c.java

---

Open Source Used In security_automation identity-guest-cs 290
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 *   https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance

* with the License. You may obtain a copy of the License at:

* https://www.apache.org/licenses/LICENSE-2.0

* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1265673536_1643922096.16/0/netty-codec-4-1-69-final-sources-
  jar/io/netty/handler/codec/compression/FastLz.java
* /opt/cola/permits/1265673536_1643922096.16/0/netty-codec-4-1-69-final-sources-
  jar/io/netty/handler/codec/compression/LzfDecoder.java
* /opt/cola/permits/1265673536_1643922096.16/0/netty-codec-4-1-69-final-sources-
  jar/io/netty/handler/codec/compression/Bzip2Decoder.java
* /opt/cola/permits/1265673536_1643922096.16/0/netty-codec-4-1-69-final-sources-
  jar/io/netty/handler/codec/compression/Bzip2BitWriter.java
* /opt/cola/permits/1265673536_1643922096.16/0/netty-codec-4-1-69-final-sources-
  jar/io/netty/handler/codec/compression/json/JsonObjectDecoder.java
* /opt/cola/permits/1265673536_1643922096.16/0/netty-codec-4-1-69-final-sources-
  jar/io/netty/handler/codec/compression/FastLzFrameDecoder.java
* /opt/cola/permits/1265673536_1643922096.16/0/netty-codec-4-1-69-final-sources-
  jar/io/netty/handler/codec/compression/FastLzFrameEncoder.java
* /opt/cola/permits/1265673536_1643922096.16/0/netty-codec-4-1-69-final-sources-
  jar/io/netty/handler/codec/compression/Bzip2Rand.java
* /opt/cola/permits/1265673536_1643922096.16/0/netty-codec-4-1-69-final-sources-
  jar/io/netty/handler/codec/compression/Bzip2Constants.java
* /opt/cola/permits/1265673536_1643922096.16/0/netty-codec-4-1-69-final-sources-
  jar/io/netty/handler/codec/compression/Crc32.java
* /opt/cola/permits/1265673536_1643922096.16/0/netty-codec-4-1-69-final-sources-
  jar/io/netty/handler/codec/compression/LzfEncoder.java
* /opt/cola/permits/1265673536_1643922096.16/0/netty-codec-4-1-69-final-sources-
  jar/io/netty/handler/codec/AsciiHeadersEncoder.java
* /opt/cola/permits/1265673536_1643922096.16/0/netty-codec-4-1-69-final-sources-
  jar/io/netty/handler/codec/compression/Bzip2HuffmanStageDecoder.java
* /opt/cola/permits/1265673536_1643922096.16/0/netty-codec-4-1-69-final-sources-
  jar/io/netty/handler/codec/compression/SnappyFramedEncoder.java
* /opt/cola/permits/1265673536_1643922096.16/0/netty-codec-4-1-69-final-sources-
  jar/io/netty/handler/codec/compression/Bzip2Encoder.java
* /opt/cola/permits/1265673536_1643922096.16/0/netty-codec-4-1-69-final-sources-
  jar/io/netty/handler/codec/compression/Bzip2MoveToFrontTable.java
* /opt/cola/permits/1265673536_1643922096.16/0/netty-codec-4-1-69-final-sources-
  jar/io/netty/handler/codec/MessageAggregationException.java
* /opt/cola/permits/1265673536_1643922096.16/0/netty-codec-4-1-69-final-sources-
  jar/io/netty/handler/codec/DecoderResultProvider.java
1.45 jsr311-api 1.1.1

1.45.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.
"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following
You must give any other recipients of the Work or Derivative Works a copy of this License; and

You must cause any modified files to carry prominent notices stating that You changed the files; and

You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may
choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and limitations under the License.
1.46 velocity 1.7

1.46.1 Available under license:

Apache Velocity

Copyright (C) 2000-2007 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).
Apache License

Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).
"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.47 slf4j-log4j 1.7.32
1.47.1 Available under license :

/**
 * Copyright (c) 2004-2011 QOS.ch
 * All rights reserved.
 *
 * Permission is hereby granted, free of charge, to any person obtaining
 * a copy of this software and associated documentation files (the
 * "Software"), to deal in the Software without restriction, including
 * without limitation the rights to use, copy, modify, merge, publish,
 * distribute, sublicense, and/or sell copies of the Software, and to
 * permit persons to whom the Software is furnished to do so, subject to
 * the following conditions:
 *
 * The above copyright notice and this permission notice shall be
 * included in all copies or substantial portions of the Software.
 *
 * THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
 * EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
 * MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
 * NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
 * LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
 * OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
 * WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
 */
1.48 jnr-constants 0.10.0

1.48.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *    http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1199743700_1630577690.39/0/jnr-constants-0-10-0-sources-1-jar/jnr/constants/Constant.java
* /opt/cola/permits/1199743700_1630577690.39/0/jnr-constants-0-10-0-sources-1-jar/jnr/constants/PlatformConstants.java
* /opt/cola/permits/1199743700_1630577690.39/0/jnr-constants-0-10-0-sources-1-jar/jnr/constants/ConstantSet.java
* /opt/cola/permits/1199743700_1630577690.39/0/jnr-constants-0-10-0-sources-1-jar/jnr/constants/platform/ConstantResolver.java

1.49 server-sdk 6.0.1.0

1.49.1 Available under license:

Found license 'Common Development and Distribution License (CDDL) 1.1' in '* The contents of this file are
subject to the terms of the * Common Development and Distribution License, Version 1.0 only * (the "License")
You may not use this file except in compliance'

1.50 jackson 1.9.2

1.50.1 Available under license:

No license file was found, but licenses were detected in source scan.

/* Jackson JSON-processor.
 *
 * Copyright (c) 2007- Tatu Saloranta, tatu.saloranta@iki.fi
 *
 * Licensed under the License specified in file LICENSE, included with
* the source code and binary code bundles.
* You may not use this file except in compliance with the License.
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1078139193_1600436032.4/0/jackson-core-asl-1-9-2-sources-jar/org/codehaus/jackson/JsonStreamContext.java
* /opt/ws_local/PERMITS_SQL/1078139193_1600436032.4/0/jackson-core-asl-1-9-2-sources-jar/org/codehaus/jackson/Base64Variant.java
* /opt/ws_local/PERMITS_SQL/1078139193_1600436032.4/0/jackson-core-asl-1-9-2-sources-jar/org/codehaus/jackson/JsonParser.java
* /opt/ws_local/PERMITS_SQL/1078139193_1600436032.4/0/jackson-core-asl-1-9-2-sources-jar/org/codehaus/jackson/JsonFactory.java
* /opt/ws_local/PERMITS_SQL/1078139193_1600436032.4/0/jackson-core-asl-1-9-2-sources-jar/org/codehaus/jackson/JsonGenerator.java
* /opt/ws_local/PERMITS_SQL/1078139193_1600436032.4/0/jackson-core-asl-1-9-2-sources-jar/org/codehaus/jackson/Base64Variants.java
* /opt/ws_local/PERMITS_SQL/1078139193_1600436032.4/0/jackson-core-asl-1-9-2-sources-jar/org/codehaus/jackson/util/ByteArrayBuilder.java

1.51 jcommander 1.78

1.51.1 Available under license:
 No license file was found, but licenses were detected in source scan.

/**
 * Copyright (C) 2010 the original author or authors.
 * See the notice.md file distributed with this work for additional
 * information regarding copyright ownership.
 * 
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 * http://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
/**

Found in path(s):
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2-jar/com/beust/jcommander/IUsageFormatter.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2-jar/com/beust/jcommander/UnixStyleUsageFormatter.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2-jar/com/beust/jcommander/JCommander.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2-jar/com/beust/jcommander/DefaultUsageFormatter.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2-jar/com/beust/jcommander/IStringConverterFactory.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2-jar/com/beust/jcommander/converters/URLConverter.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2-jar/com/beust/jcommander/internal/Lists.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2-jar/com/beust/jcommander/ParameterException.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2-jar/com/beust/jcommander/internal/DefaultConverterFactory.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2-jar/com/beust/jcommander/converters/IntegerConverter.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2-jar/com/beust/jcommander/converters/FileConverter.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2-jar/com/beust/jcommander/converters/StringConverter.java

No license file was found, but licenses were detected in source scan.

/ **
* Copyright (C) 2010 the original author or authors.
* See the notice.md file distributed with this work for additional
* information regarding copyright ownership.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2-jar/com/beust/jcommander/IStringConverterFactory.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2-jar/com/beust/jcommander/converters/URLConverter.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2-jar/com/beust/jcommander/internal/Lists.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2-jar/com/beust/jcommander/ParameterException.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2-jar/com/beust/jcommander/internal/DefaultConverterFactory.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2-jar/com/beust/jcommander/converters/IntegerConverter.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2-jar/com/beust/jcommander/converters/FileConverter.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2-jar/com/beust/jcommander/converters/StringConverter.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2.jar/com/beust/jcommander/IStringConverter.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2.jar/com/beust/jcommander/internal/Sets.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2.jar/com/beust/jcommander/converters/LongConverter.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2.jar/com/beust/jcommander/converters/InetAddressConverter.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2.jar/com/beust/jcommander/IDefaultProvider.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2.jar/com/beust/jcommander/converters/DoubleConverter.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2.jar/com/beust/jcommander/converters/BigDecimalConverter.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2.jar/com/beust/jcommander/converters/FloatConverter.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2.jar/com/beust/jcommander/converters/ISO8601DateConverter.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2.jar/com/beust/jcommander/MissingCommandException.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2.jar/com/beust/jcommander/converters/NoConverter.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2.jar/com/beust/jcommander/converters/BooleanConverter.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2.jar/com/beust/jcommander/converters/Parameter.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2.jar/com/beust/jcommander/ParametersDelegate.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2.jar/com/beust/jcommander/Parameters.java

No license file was found, but licenses were detected in source scan.
/**
 * Copyright (C) 2011 the original author or authors.
 * See the notice.md file distributed with this work for additional
 * information regarding copyright ownership.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2-
jar/com/beust/jcommander/validators/NoValidator.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2-
jar/com/beust/jcommander/validators/PositiveInteger.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2-
jar/com/beust/jcommander/validators/NoValueValidator.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2-
jar/com/beust/jcommander/IParameterValidator.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2-
jar/com/beust/jcommander/IParameterValidator2.java

### 1.52 jctools-core 3.1.0

**1.52.1 Available under license:**
No license file was found, but licenses were detected in source scan.

/*
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/0/jctools-core-3-1-0-sources-jar/org/jctools/queues/spec/Ordering.java
* /opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/0/jctools-core-3-1-0-sources-jar/org/jctools/queues/atomic/SpscUnboundedAtomicArrayQueue.java
* /opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/0/jctools-core-3-1-0-sources-jar/org/jctools/queues/atomic/SpmcAtomicArrayQueue.java
* /opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/0/jctools-core-3-1-0-sources-jar/org/jctools/queues/MessagePassingQueueUtil.java
* /opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/0/jctools-core-3-1-0-sources-jar/org/jctools/queues/atomic/MpmcAtomicArrayQueue.java
* /opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/0/jctools-core-3-1-0-sources-jar/org/jctools/queues/util/UnsafeJvmInfo.java
* /opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/0/jctools-core-3-1-0-sources-jar/org/jctools/queues/atomic/BaseSpscLinkedAtomicArrayQueue.java
* /opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/0/jctools-core-3-1-0-sources-jar/org/jctools/queues/atomic/AtomicQueueFactory.java
* /opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/0/jctools-core-3-1-0-sources-jar/org/jctools/queues/MpscGrowableArrayQueue.java
* /opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/0/jctools-core-3-1-0-sources-jar/org/jctools/queues/atomic/BaseSpscLinkedAtomicArrayQueue.java
* /opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/0/jctools-core-3-1-0-sources-jar/org/jctools/queues/atomic/AtomicQueueFactory.java
* /opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/0/jctools-core-3-1-0-sources-jar/org/jctools/queues/MpmcArrayQueue.java
jar/org/jctools/queues/MpscUnboundedXaddChunk.java
* /opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/0/jctools-core-3-1-0-sources-
jar/org/jctools/queues/MpscChinkedArrayQueue.java
* /opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/0/jctools-core-3-1-0-sources-
jar/org/jctools/maps/NonBlockingHashMapLong.java
* /opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/0/jctools-core-3-1-0-sources-
jar/org/jctools/queues/atomic/MpscChinkedAtomicArrayQueue.java
* /opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/0/jctools-core-3-1-0-sources-
jar/org/jctools/queues/ConcurrentSequencedCircularArrayQueue.java
* /opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/0/jctools-core-3-1-0-sources-
jar/org/jctools/util/UnsafeLongArrayAccess.java
* /opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/0/jctools-core-3-1-0-sources-
jar/org/jctools/queues/MpscBlockingConsumerArrayQueue.java
* /opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/0/jctools-core-3-1-0-sources-
jar/org/jctools/queues/atomic/BaseMpscLinkedAtomicArrayQueue.java
* /opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/0/jctools-core-3-1-0-sources-
jar/org/jctools/maps/NonBlockingHashMap.java
* /opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/0/jctools-core-3-1-0-sources-
jar/org/jctools/queues/BaseMpscLinkedArrayQueue.java
* /opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/0/jctools-core-3-1-0-sources-
jar/org/jctools/queues/atomic/MpscGrowableAtomicArrayQueue.java
* /opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/0/jctools-core-3-1-0-sources-
jar/org/jctools/maps/NonBlockingIdentityHashMap.java
* /opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/0/jctools-core-3-1-0-sources-
jar/org/jctools/queues/BaseLinkedQueue.java
* /opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/0/jctools-core-3-1-0-sources-
jar/org/jctools/util/InternalAPI.java
* /opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/0/jctools-core-3-1-0-sources-
jar/org/jctools/maps/NonBlockingHashSet.java
* /opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/0/jctools-core-3-1-0-sources-
jar/org/jctools/util/Pow2.java
* /opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/0/jctools-core-3-1-0-sources-
jar/org/jctools/queues/spec/ConcurrentQueueSpec.java
* /opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/0/jctools-core-3-1-0-sources-
jar/org/jctools/queues/atomic/SpscChinkedAtomicArrayQueue.java
* /opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/0/jctools-core-3-1-0-sources-
jar/org/jctools/maps/ConcurrentAutoTable.java
* /opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/0/jctools-core-3-1-0-sources-
jar/org/jctools/queues/MpscUnboundedXaddChunk.java
* /opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/0/jctools-core-3-1-0-sources-
jar/org/jctools/queues/atomic/MpscUnboundedAtomicArrayQueue.java
* /opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/0/jctools-core-3-1-0-sources-
jar/org/jctools/queues/LinkedQueueNode.java
* /opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/0/jctools-core-3-1-0-sources-
jar/org/jctools/queues/LinkedQueueNode.java
* /opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/0/jctools-core-3-1-0-sources-
jar/org/jctools/queues/SpacUnboundedArrayQueue.java
* /opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/0/jctools-core-3-1-0-sources-
jar/org/jctools/util/PortableJvmInfo.java
* /opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/0/jctools-core-3-1-0-sources-
No license file was found, but licenses were detected in source scan.

<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/maven-v4_0_0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <artifactId>jctools-core</artifactId>
</project>
<groupId>org.jctools</groupId>
<version>3.1.0</version>
<name>Java Concurrency Tools Core Library</name>
<description>Java Concurrency Tools Core Library</description>
<packaging>bundle</packaging>

<dependencies>
<dependency>
<groupId>org.hamcrest</groupId>
<artifactId>hamcrest-all</artifactId>
<version>${hamcrest.version}</version>
<scope>test</scope>
</dependency>

<dependency>
<groupId>junit</groupId>
<artifactId>junit</artifactId>
<version>${junit.version}</version>
<scope>test</scope>
</dependency>

<dependency>
<groupId>com.google.guava</groupId>
<artifactId>guava-testlib</artifactId>
<version>${guava-testlib.version}</version>
<scope>test</scope>
</dependency>
</dependencies>

<build>
<plugins>
<plugin>
<groupId>org.apache.maven.plugins</groupId>
<artifactId>maven-surefire-plugin</artifactId>
<version>3.0.0-M3</version>
<configuration>
<includes>
<include>*</include>
</includes>
</configuration>
</plugin>
<plugin>
<groupId>org.apache.felix</groupId>
<artifactId>maven-bundle-plugin</artifactId>
<version>4.2.1</version>
<extensions>true</extensions>
<configuration>
<instructions>
<Import-Package>sun.misc;resolution=optional</Import-Package>
</configuration>
</plugin>
</plugins>
</build>
<instructions>
</instructions>
</configuration>
</plugin>
<plugin>
<groupId>org.apache.maven.plugins</groupId>
<artifactId>maven-source-plugin</artifactId>
<version>3.2.0</version>
<executions>
<execution>
<id>attach-sources</id>
<phase>verify</phase>
<goals>
<goal>jar-no-fork</goal>
</goals>
</execution>
</executions>
</plugin>
<plugin>
<groupId>org.apache.maven.plugins</groupId>
<artifactId>maven-javadoc-plugin</artifactId>
<version>3.1.1</version>
<configuration>
<additionalOptions>
<Xdoclint:none</additionalOptions>
</additionalOptions>
<source>8</source>
</configuration>
<executions>
<execution>
<id>attach-javadocs</id>
<goals>
<goal>jar</goal>
</goals>
</execution>
</executions>
</plugin>
</plugins>
</build>

<distributionManagement>
<repository>
<id>bintray-jctools-jctools</id>
<name>jctools-jctools</name>
:url>https://api.bintray.com/maven/jctools/jctools/jctools-core;publish=1</url>
</repository>
</distributionManagement>

<url>https://bintray.com/maven/jctools/jctools/jctools-core</url>

<url>https://github.com/JCTools</url>
<hamcrest.version>1.3</hamcrest.version>
<junit.version>4.12</junit.version>
<guava-testlib.version>21.0</guava-testlib.version>
</properties>
</project>

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/0/jctools-core-3-1-0-sources-jar/META-INF/maven/org.jctools/jctools-core/pom.xml

1.53 httpclient-cache 4.5.13

1.53.1 Available under license:

Apache HttpClient Cache
Copyright 2010-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation
"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

   You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.54 spring-rabbit 2.3.11

1.54.1 Available under license:
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015-2021 the original author or authors.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * https://www.apache.org/licenses/LICENSE-2.0
 * Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/listener/adapter/HandlerAdapter.java
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/listener/MultiMethodRabbitListenerEndpoint.java
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/listener/adapter/DelegatingInvocableHandler.java

No license file was found, but licenses were detected in source scan.

/ *
* Copyright 2017-2019 the original author or authors.
* *
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* *
* https://www.apache.org/licenses/LICENSE-2.0
* *
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/connection/ConnectionBlockedEvent.java
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/connection/ShutDownChannelListener.java
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/connection/ConnectionUnblockedEvent.java
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/listener/ConsumeOkEvent.java
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/core/CorrelationDataPostProcessor.java

No license file was found, but licenses were detected in source scan.

/ *
* Copyright 2014-2019 the original author or authors.
* *
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
*      https://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-
  jar/org/springframework/amqp/rabbit/Listener/RabbitListenerEndpointRegistry.java
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-
  jar/org/springframework/amqp/rabbit/support/ValueExpression.java
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-
  jar/org/springframework/amqp/rabbit/annotation/EnableRabbit.java
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-
  jar/org/springframework/amqp/rabbit/config/RetryInterceptorBuilder.java
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-
  jar/org/springframework/amqp/rabbit/config/SimpleRabbitListenerEndpoint.java
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-
  jar/org/springframework/amqp/rabbit/connection/SimpleRoutingConnectionFactory.java
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-
  jar/org/springframework/amqp/rabbit/listener/QueuesNotAvailableException.java
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-
  jar/org/springframework/amqp/rabbit/listener/adapter/ReplyFailureException.java
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-
  jar/org/springframework/amqp/rabbit/listener/FatalExceptionStrategy.java
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-
  jar/org/springframework/amqp/rabbit/core/BatchingRabbitTemplate.java
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-
  jar/org/springframework/amqp/rabbit/batch/BatchingStrategy.java
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-
  jar/org/springframework/amqp/rabbit/connection/SimpleResourceHolder.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2019-2021 the original author or authors.
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
Open Source Used In security_automation identity-guest-cs 320

* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/annotation/RabbitListenerConfigurationSelector.java

No license file was found, but licenses were detected in source scan.

/ *
* Copyright 2019 the original author or authors.
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
*   https://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/retry/MessageBatchRecoverer.java
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/config/ContainerCustomizer.java
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/connection/PublisherCallbackConnectionFactory.java
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/listener/adapter/BatchMessagingMessageListenerAdapter.java
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/listener/api/ChannelAwareBatchMessagingMessageListener.java

No license file was found, but licenses were detected in source scan.

/ *
* Copyright 2020 the original author or authors.
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/connection/ConnectionFactoryContextWrapper.java
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/connection/PooledChannelConnectionFactory.java
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/listener/adapter/ReplyPostProcessor.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2002-2019 the original author or authors.
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/core/RabbitGatewaySupport.java
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/annotation/RabbitListenerConfigurer.java
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/config/AbstractRetryOperationsInterceptorFactoryBean.java
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/config/QueueParser.java
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/listener/exception/FatalListenerStartupException.java
* /opt/cola/permits/1220178530_1635319508.30/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/connection/ConsumerChannelRegistry.java
* /opt/cola/permits/1220178530_1635319508.30/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/support/RabbitExceptionTranslator.java
* /opt/cola/permits/1220178530_1635319508.30/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/connection/SimpleConnection.java
* /opt/cola/permits/1220178530_1635319508.30/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/config/StatefulRetryOperationsInterceptorFactoryBean.java
* /opt/cola/permits/1220178530_1635319508.30/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/config/AnnotationDrivenParser.java
* /opt/cola/permits/1220178530_1635319508.30/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/connection/ConnectionFactory.java
* /opt/cola/permits/1220178530_1635319508.30/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/connection/ConnectionProxy.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2018-2020 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *      https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
No license file was found, but licenses were detected in source scan.

No license file was found, but licenses were detected in source scan.

No license file was found, but licenses were detected in source scan.
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014-2020 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *      https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/listener/AbstractRabbitListenerEndpoint.java
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/listener/MethodRabbitListenerEndpoint.java
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/listener/ConditionalRejectingErrorHandler.java
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/config/SimpleRabbitListenerContainerFactory.java
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/listener/ListenerContainerConsumerFailedEvent.java
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/listener/ListenerContainerConsumerTerminatedEvent.java
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/annotation/RabbitListener.java
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/annotation/QueueBinding.java
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/annotation/Queue.java
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/listener/ListenerContainerConsumerFailedEvent.java
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/listener/ListenerContainerConsumerTerminatedEvent.java
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/annotation/RabbitHandler.java
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/connection/RoutingConnectionFactory.java
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/connection/LocalizedQueueConnectionFactory.java

No license file was found, but licenses were detected in source scan.
/*
* Copyright 2002-2020 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/annotation/RabbitBootstrapConfiguration.java
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/core/RabbitOperations.java
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/connection/RabbitUtils.java
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/connection/AbstractRoutingConnectionFactory.java
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/connection/PublisherCallbackChannel.java
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/connection/PendingConfirm.java
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/connection/RabbitConnectionFactoryBean.java
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/connection/RabbitListenerEndpoint.java
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/connection/PublisherCallbackChannel.java
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/connection/PendingConfirm.java
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/connection/RabbitConnectionFactoryBean.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2002-2021 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*/
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

No license file was found, but licenses were detected in source scan.
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2016-2021 the original author or authors.
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
*   https://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2016-2019 the original author or authors.
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
*   https://www.apache.org/licenses/LICENSE-2.0
*/
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/core/RabbitAdminEvent.java
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/connection/AutoRecoverConnectionNotCurrentlyOpenException.java
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/connection/ConnectionNameStrategy.java
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/listener/DirectReplyToMessageListenerContainer.java
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/support/Delivery.java
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/listener/AsyncConsumerStartedEvent.java
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/listener/ListenerFailedRuleBasedTransactionAttribute.java
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/listener/AsyncConsumerRestartedEvent.java
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/listener/AsyncConsumerStoppedEvent.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2016-2020 the original author or authors.
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

*/

Found in path(s):
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/log4j2/AmqpAppender.java
No license file was found, but licenses were detected in source scan.

*/

* Copyright 2020-2021 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at
*

https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/connection/ThreadChannelConnectionFactory.java
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/annotation/MultiRabbitListenerAnnotationBeanPostProcessor.java
No license file was found, but licenses were detected in source scan.

*/

* Copyright 2014-2021 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/listener/adapter/AbstractAdaptableMessageListener.java
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/listener/RabbitListenerEndpointRegistrar.java
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/annotation/RabbitListenerAnnotationBeanPostProcessor.java
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/config/AbstractRabbitListenerContainerFactory.java
* /opt/cola/permits/1220178530_1635319508.3/0/spring-rabbit-2-3-11-sources-jar/org/springframework/amqp/rabbit/retry/RepublishMessageRecoverer.java

1.55 lz4-java 1.7.1
1.55.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.56 jackson-module-jsonschema 2.13.0

1.56.1 Available under license:

This copy of Jackson JSON processor databind module is licensed under the
Apache (Software) License, version 2.0 ("the License").
See the License for details about distribution rights, and the
specific rights regarding derivate works.
You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

1.57 zstd 1.4.5

1.57.1 Available under license:

BSD License

For Zstandard software

Copyright (c) 2016-present, Facebook, Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name Facebook nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

GNU GENERAL PUBLIC LICENSE

Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble
The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software—to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Lesser General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.
GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License: they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a
notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)
The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this
License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free
programs whose distribution conditions are different, write to the author
to ask for permission. For software which is copyrighted by the Free
Software Foundation, write to the Free Software Foundation; we sometimes
make exceptions for this. Our decision will be guided by the two goals
of preserving the free status of all derivatives of our free software and
of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY
FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN
OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES
PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED
OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS
TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE
PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING,
REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING
WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR
REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES,
INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING
OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED
TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY
YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER
PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE
POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest
possible use to the public, the best way to achieve this is to make it
free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest
to attach them to the start of each source file to most effectively
convey the exclusion of warranty; and each file should have at least
the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License.

1.58 activation-api 1.1
1.58.1 Available under license:

# Notices for Eclipse Project for JAF

This content is produced and maintained by the Eclipse Project for JAF project.

* Project home: https://projects.eclipse.org/projects/ee4j.jaf

## Copyright

All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed source code repository logs.

## Declared Project Licenses

This program and the accompanying materials are made available under the terms of the Eclipse Distribution License v. 1.0, which is available at http://www.eclipse.org/org/documents/edl-v10.php.

SPDX-License-Identifier: BSD-3-Clause

## Source Code

The project maintains the following source code repositories:

* https://github.com/eclipse-ee4j/jaf

## Third-party Content

This project leverages the following third party content.

JUnit (4.12)

* License: Eclipse Public License

Copyright (c) 2018 Oracle and/or its affiliates. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of the Eclipse Foundation, Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.59 archaius-core 0.4.1

1.60 dom 1.0
1.60.1 Available under license :
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2001 World Wide Web Consortium,
 * (Massachusetts Institute of Technology, Institut National de
 * Recherche en Informatique et en Automatique, Keio University). All
 * Rights Reserved. This program is distributed under the W3C's Software
 * Intellectual Property License. This program is distributed in the
 * hope that it will be useful, but WITHOUT ANY WARRANTY; without even
 * the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR
 * PURPOSE.
 * See W3C License http://www.w3.org/Consortium/Legal/ for more details.
 */

Found in path(s):
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/dom3/as/ElementEditAS.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/dom3/as/ASNotationDeclaration.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/dom3/as/ASObject.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/dom3/as/CharacterDataEditAS.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/dom3/as/DOMImplementationAS.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/dom3/as/ASModel.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/dom3/as/DocumentAS.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/dom3/as/ASAttributeDeclaration.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/dom3/as/ASElementDeclaration.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/dom3/as/ASEntityDeclaration.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/dom3/as/ASNamedObjectMap.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/dom3/as/ASDataType.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/dom3/as/ASContentModel.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/dom3/as/DOMASWriter.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/dom3/as/DocumentEditAS.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/dom3/as/AObjectList.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/dom3/as/DOMASBuilder.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/dom3/as/DOMASException.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/dom3/as/NodeEditAS.java

No license file was found, but licenses were detected in source scan.

Licensed to the Apache Software Foundation (ASF) under one or more
contributor license agreements. See the NOTICE file distributed with
this work for additional information regarding copyright ownership.
The ASF licenses this file to You under the Apache License, Version 2.0
(the "License"); you may not use this file except in compliance with
the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE
2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Found in path(s):
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2000 World Wide Web Consortium,
 * (Massachusetts Institute of Technology, Institut National de
 * Recherche en Informatique et en Automatique, Keio University). All
 * Rights Reserved. This program is distributed under the W3C's Software
 * Intellectual Property License. This program is distributed in the
 * hope that it will be useful, but WITHOUT ANY WARRANTY; without even
 * the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR
 * PURPOSE. See W3C License http://www.w3.org/Consortium/Legal/ for more
 * details.
 */

Found in path(s):
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/xx/datatypes/package.html

No license file was found, but licenses were detected in source scan.

/*
 * Licensed to the Apache Software Foundation (ASF) under one or more
 * contributor license agreements. See the NOTICE file distributed with
 * this work for additional information regarding copyright ownership.
 * The ASF licenses this file to You under the Apache License, Version 2.0
 * (the "License"); you may not use this file except in compliance with
 * the License. You may obtain a copy of the License at
 * *
 *      http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

/**
 * Checks if this content model has had its min/maxOccurs values reduced for
 * purposes of speeding up UPA. If so, this content model should not be used
 * for any purpose other than checking unique particle attribution
 * *
 * @return a boolean that says whether this content has been compacted for UPA
 */

Found in path(s):
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/impl/xx/models/XSCMValidator.java
No license file was found, but licenses were detected in source scan.

/*
 * Licensed to the Apache Software Foundation (ASF) under one or more
 * contributor license agreements. See the NOTICE file distributed with
 * this work for additional information regarding copyright ownership.
 * The ASF licenses this file to You under the Apache License, Version 2.0
 * (the "License"); you may not use this file except in compliance with
 * the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

/**
* DOM Level 3 LS CR - Experimental.
* Create a new <code>LSParser</code>. The newly constructed parser may
* then be configured by means of its <code>DOMConfiguration</code>
* object, and used to parse documents by means of its <code>parse</code>
* method.
* @param mode  The <code>mode</code> argument is either
*   <code>MODE_SYNCHRONOUS</code> or <code>MODE_ASYNCHRONOUS</code>, if
*   <code>mode</code> is <code>MODE_SYNCHRONOUS</code> then the
*   <code>LSParser</code> that is created will operate in synchronous
*   mode, if it's <code>MODE_ASYNCHRONOUS</code> then the
*   <code>LSParser</code> that is created will operate in asynchronous
*   mode.
* @param schemaType  An absolute URI representing the type of the schema
*   language used during the load of a <code>Document</code> using the
*   newly created <code>LSParser</code>. Note that no lexical checking
*   is done on the absolute URI. In order to create a
*   <code>LSParser</code> for any kind of schema types (i.e. the
*   LSParser will be free to use any schema found), use the value
*   <code>null</code>.
* @return  The newly created <code>LSParser</code> object. This
*<p><b>Note:</b> For W3C XML Schema [<a href='http://www.w3.org/TR/2001/REC-xmlschema-1-20010502/'>XML Schema Part 1</a>]
*   , applications must use the value
*   <code>"http://www.w3.org/2001/XMLSchema"</code>. For XML DTD [<a href='http://www.w3.org/TR/2000/REC-xml-20001006'>XML 1.0</a>],
*   applications must use the value
*   <code>"http://www.w3.org/TR/REC-xml"</code>. Other Schema languages
*   are outside the scope of the W3C and therefore should recommend an
*   absolute URI in order to use this method.
* @return  The newly created <code>LSParser</code> object. This
* <code>LSParser</code> is either synchronous or asynchronous depending on the value of the <code>mode</code> argument.

* <p>Note:</p> By default, the newly created <code>LSParser</code> does not contain a <code>DOMErrorHandler</code>, i.e. the value of the "<a href='http://www.w3.org/TR/2003/WD-DOM-Level-3-Core-20030609/core.html#parameter-error-handler'>error-handler</a>" configuration parameter is <code>null</code>. However, implementations may provide a default error handler at creation time. In that case, the initial value of the <code>error-handler</code> configuration parameter on the new created <code>LSParser</code> contains a reference to the default error handler.

* @exception DOMException

* NOT_SUPPORTED_ERR: Raised if the requested mode or schema type is not supported.

/*

Found in path(s):
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/dom/CoreDOMImplementationImpl.java
No license file was found, but licenses were detected in source scan.

/*

* Licensed to the Apache Software Foundation (ASF) under one or more contributor license agreements. See the NOTICE file distributed with this work for additional information regarding copyright ownership.
* The ASF licenses this file to You under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
*
/*

Found in path(s):
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/dom/DOMImplementationListImpl.java
No license file was found, but licenses were detected in source scan.

/*

* Licensed to the Apache Software Foundation (ASF) under one or more contributor license agreements. See the NOTICE file distributed with this work for additional information regarding copyright ownership.
* The ASF licenses this file to You under the Apache License, Version 2.0
* (the "License"); you may not use this file except in compliance with
* the License. You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
/**
* used to check the 3 constraints against each complex type
* (should be each model group):
* Unique Particle Attribution, Particle Derivation (Restriction),
* Element Declrations Consistent.
*/

Found in path(s):
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
  jar/org/apache/xerces/impl/xs/XSConstraints.java

No license file was found, but licenses were detected in source scan.

/*
* Licensed to the Apache Software Foundation (ASF) under one or more
* contributor license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright ownership.
* The ASF licenses this file to You under the Apache License, Version 2.0
* (the "License"); you may not use this file except in compliance with
* the License. You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
  jar/org/apache/xerces/impl/xs/traversers/XSDocumentInfo.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
  jar/org/apache/xerces/impl/xs/traversers/XSDElementTraverser.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
  jar/org/apache/xerces/xni/parser/XMLErrorHandler.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
Open Source Used In security_automation identity-guest-cs 361
Open Source Used in security_automation identity-guest-cs 363
Open Source Used In security_automation identity-guest-cs 364
jar/org/apache/xerces/dom/PSVIDocumentImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/impl/dtd/DTDGrammarBucket.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/impl/xs/ElementPSVImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/impl/xpath/regex/REUtil.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/impl/xpath/regex/Op.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/xni/parser/XMLParseException.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/impl/xpath/regex/Op.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/impl/dv/dtd/NMTOKENDatatypeValidator.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/html/dom/HTMLParagraphElementImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/dom/DOMImplementationSourceImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/impl/dv/dtd/NMTOKENDatatypeValidator.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/html/dom/HTMLParagraphElementImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/impl/dv/xs/ListDV.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/impl/dv/dtd/NMTOKENDatatypeValidator.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/html/dom/HTMLULListElementImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/wml/WMLBrElement.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/impl/util/SymbolHash.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/impl/dv/xs/ListDV.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/impl/dv/dtd/NMTOKENDatatypeValidator.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/dom/CharacterDataImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/dom/RangeImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/dom/CharacterDataImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/dom/RangeImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xml/serialize/SerializerFactoryImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/wml/WMLBrElement.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/dom/CharacterDataImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/dom/RangeImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/dom/CharacterDataImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/dom/RangeImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/dom/CharacterDataImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/dom/RangeImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/dom/CharacterDataImpl.java
Open Source Used In security_automation identity-guest-cs 370

- /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/html/dom/HTMLOptionElementImpl.java
- /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/wml/dom/WMLTrElementImpl.java
- /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/html/dom/HTMLDOMImplementationImpl.java
- /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/impl/xs/models/XSCMUniOp.java
- /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/xni/grammars/XMLGrammarLoader.java
- /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/impl/validation/ConfigurableValidationState.java
- /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xml/serialize/IndentPrinter.java
- /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xml/serialize/ElementState.java
- /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xml/serialize/ObjectFactory.java
- /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/dom/PSVIDOMImplementationImpl.java
- /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/impl/dtd/models/DFAContentModel.java
- /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/impl/dtd/AttributeMap.java
- /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/stax/events/EndDocumentImpl.java
- /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/wml/dom/WMLPElementImpl.java
- /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/xni/grammars/Grammar.java
- /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/impl/xpath/regex/Match.java
- /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/impl/xs/XSMessageFormatter.java
- /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/impl/xpath/regex/Match.java
- /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/wml/dom/WMLAccessElementImpl.java
- /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/xni/grammars/Grammar.java
Open Source Used In security_automation identity-guest-cs 371
jar/org/apache/html/dom/HTMLHeadElementImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/impl/xs/models/XSCMLeaf.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/impl/xx/identity/Selector.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/impl/dv/dtd/XML11IDDatatypeValidator.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/xs/XSElementDeclaration.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/impl/xs/XSAnnotationImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/impl/xx/identity/Selector.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/impl/xpath/XPathException.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/stax/events/EntityDeclarationImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/parsers/AbstractSAXParser.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/wml/dom/WMLBigElementImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/xni/XMLLocator.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/xs/XSTypeDefinition.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/impl/xx/ItemPSVI.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/impl/xx/URI.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/html/dom/HTMLBuilder.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/jaxp/validation/WeakReferenceXMLSchema.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/xs/datatypes/XSQName.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xercesdom/NodeListImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/impl/dv/xs/AnyAtomicDV.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/impl/dv/xx/AnyAtomicDV.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/xs/ItemPSVI.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/impl/xx/NamedNodeMapImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/html/dom/HTMLObjectElementImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/impl/xx/DefaultErrorHandler.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/parsers/DTDConfiguration.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/jaxp/validation/SimpleXMLSchema.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/html/dom/HTMLLegendElementImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/impl/dv/xs/YearMonthDurationDV.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/impl/xs/traversers/XSDWildcardTraverser.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/html/dom/HTMLHeadingElementImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/impl/dv/XSFacets.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/impl/xs/ElementPSVI.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/wml/WMLRefreshElement.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/impl/xs/util/XSGrammarPool.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/xni/XMLResourceIdentifier.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/html/dom/HTMLDirectoryElementImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/xni/XMLDTDHandler.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/xs/XSMultiValueFacet.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/xs/XSAnnotation.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/html/dom/HTMLLinkElementImpl.java

Open Source Used In security_automation identity-guest-cs 373
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/dom/events/EventImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/impl/dv/DTDDefinitionsFactory.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/impl/xs/identity/ValueStore.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/impl/xs/XSElementDecl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/xs/PSVIProvider.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/impl/xs/XMLSchemaException.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/xs/XSNamedMap.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/impl/xs/XSAttributeUseImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/impl/xs/xs/XSElementDecl.java

Open Source Used In security_automation identity-guest-cs 374
No license file was found, but licenses were detected in source scan.

/*
 * Licensed to the Apache Software Foundation (ASF) under one or more
 * contributor license agreements. See the NOTICE file distributed with
 * this work for additional information regarding copyright ownership.
 * The ASF licenses this file to You under the Apache License, Version 2.0
 * (the "License"); you may not use this file except in compliance with
 * the License. You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
  jar/org/apache/xerces/impl/dv/xs/AbstractDateTimeDV.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
  jar/org/apache/xerces/impl/dv/xs/YearDV.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
  jar/org/apache/xerces/impl/xs/traversers/XSDKeyrefTraverser.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
  jar/org/apache/xerces/impl/dv/xs/YearMonthDV.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
  jar/org/apache/xerces/impl/dv/xs/DateDV.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
  jar/org/apache/xerces/impl/dv/xs/TimeDV.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
  jar/org/apache/xerces/impl/dv/xs/DateTimeDV.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
  jar/org/apache/xerces/impl/dv/xs/XSSimpleTypeDecl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
  jar/org/apache/xerces/impl/dv/xs/DayDV.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
  jar/org/apache/xerces/impl/dv/xs/DurationDV.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
  jar/org/apache/xerces/impl/xs/XMLSchemaValidator.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
  jar/org/apache/xerces/impl/xs/traversers/XSDHandler.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
  jar/org/apache/xerces/impl/xs/traversers/XSAttributeChecker.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
  jar/org/apache/xerces/impl/xs/XSModelGroupImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
  jar/org/apache/xerces/impl/xs/XSSimpleTypeDecl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
  jar/org/apache/xerces/impl/xs/DayDV.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
  jar/org/apache/xerces/impl/xs/XSDUniqueOrKeyTraverser.java

No license file was found, but licenses were detected in source scan.

# The ASF licenses this file to You under the Apache License, Version 2.0
# (the "License"); you may not use this file except in compliance with
# the License. You may obtain a copy of the License at
# http://www.apache.org/licenses/LICENSE-2.0
# distributed under the License is distributed on an "AS IS" BASIS,
No license file was found, but licenses were detected in source scan.

/*
 * Licensed to the Apache Software Foundation (ASF) under one or more
 * contributor license agreements. See the NOTICE file distributed with
 * this work for additional information regarding copyright ownership.
 * The ASF licenses this file to You under the Apache License, Version 2.0
 * (the "License"); you may not use this file except in compliance with
 * the License. You may obtain a copy of the License at
 *
 *      http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

// Unique Particle Attribution

Found in path(s):
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/impl/xs/models/XSDFACM.java
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).
"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed.
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

=======================================================================

SPRING FRAMEWORK 5.3.12 SUBCOMPONENTS:

Spring Framework 5.3.12 includes a number of subcomponents with separate copyright notices and license terms. The product that includes this file does not necessarily use all the open source subcomponents referred to below. Your use of the source code for these subcomponents is subject to the terms and conditions of the following licenses.


Copyright (c) 2000-2011 INRIA, France Telecom
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of the copyright holders nor the names of its
contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright (c) 1999-2009, OW2 Consortium <https://www.ow2.org/>

>>> CGLIB 3.3 (cglib:cglib:3.3):

Per the LICENSE file in the CGLIB JAR distribution downloaded from https://github.com/cglib/cglib/releases/download/RELEASE_3_3_0/cglib-3.3.0.jar, CGLIB 3.3 is licensed under the Apache License, version 2.0, the text of which is included above.

>>> Objenesis 3.1 (org.objenesis:objenesis:3.1):

Per the LICENSE file in the Objenesis ZIP distribution downloaded from http://objenesis.org/download.html, Objenesis 3.1 is licensed under the Apache License, version 2.0, the text of which is included above.

Per the NOTICE file in the Objenesis ZIP distribution downloaded from http://objenesis.org/download.html and corresponding to section 4d of the Apache License, Version 2.0, in this case for Objenesis:

Objenesis
Copyright 2006-2019 Joe Walnes, Henri Tremblay, Leonardo Mesquita

=================================================================================

To the extent any open source components are licensed under the EPL and/or other similar licenses that require the source code and/or modifications to source code to be made available (as would be noted above), you may obtain a copy of the source code corresponding to the binaries for such open source components and modifications thereto, if any, (the "Source Files"), by downloading the Source Files from https://spring.io/projects, Pivotal's website.
at https://network.pivotal.io/open-source, or by sending a request, with your name and address to: Pivotal Software, Inc., 875 Howard Street, 5th floor, San Francisco, CA 94103, Attention: General Counsel. All such requests should clearly specify: OPEN SOURCE FILES REQUEST, Attention General Counsel. Pivotal can mail a copy of the Source Files to you on a CD or equivalent physical medium.

This offer to obtain a copy of the Source Files is valid for three years from the date you acquired this Software product. Alternatively, the Source Files may accompany the Software.

Spring Framework 5.3.12
Copyright (c) 2002-2021 Pivotal, Inc.

This product is licensed to you under the Apache License, Version 2.0 (the "License"). You may not use this product except in compliance with the License.

This product may include a number of subcomponents with separate copyright notices and license terms. Your use of the source code for these subcomponents is subject to the terms and conditions of the subcomponent's license, as noted in the license.txt file.

1.63 commons-collections 4.4

1.63.1 Available under license:
Apache Commons Collections
Copyright 2001-2019 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.
2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.64 commons-pool 2.9.0
1.64.1 Available under license:
Apache Commons Pool
Copyright 2001-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (https://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License. You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

APACHE COMMONS POOL DERIVATIVE WORKS:

The LinkedBlockingDeque implementation is based on an implementation written by
Doug Lea with assistance from members of JCP JSR-166 Expert Group and released
to the public domain, as explained at
http://creativecommons.org/licenses/publicdomain

1.65 jersey-multipart 1.19.1
1.65.1 Available under license:

Found license 'General Public License 2.0' in * Copyright (c) 2010-2012 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You * may not use this file except in compliance with the License. You can * Oracle designates this particular file as subject to the "Classpath" * exception as provided by Oracle in the GPL Version 2 section of the License’
Found license 'General Public License 2.0' in * Copyright (c) 2010-2011 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You * may not use this file except in compliance with the License. You can * Oracle designates this particular file as subject to the "Classpath" * exception as provided by Oracle in the GPL Version 2 section of the License’
Found license 'General Public License 2.0' in * Copyright (c) 2013 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You * may not use this file except in compliance with the License. You can * Oracle designates this particular file as subject to the "Classpath" * exception as provided by Oracle in the GPL Version 2 section of the License’
Found license 'General Public License 2.0' in * Copyright (c) 2010-2013 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You * may not use this file except in compliance with the License. You can * Oracle designates this particular file as subject to the "Classpath" * exception as provided by Oracle in the GPL Version 2 section of the License’
1.66 spring-expression 5.3.12

1.66.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2016 the original author or authors.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * https://www.apache.org/licenses/LICENSE-2.0
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-jar/org/springframework/expression/spel/support/ReflectivePropertyAccessor.java
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-jar/org/springframework/expression/spel/ast/MethodReference.java
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-jar/org/springframework/expression/spel/standard/SpelExpression.java
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-jar/org/springframework/expression/spel/ast/StandardTypeLocator.java
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-jar/org/springframework/expression/spel/support/StandardTypeLocator.java
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-jar/org/springframework/expression/spel/standard/SpelCompiler.java
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-jar/org/springframework/expression/spel/ast/AstUtils.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2016 the original author or authors.
 */
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*  
*      https://www.apache.org/licenses/LICENSE-2.0
*  
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-
  jar/org/springframework/expression/spel/standard/TokenKind.java
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-
  jar/org/springframework/expression/ConstructorResolver.java
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-
  jar/org/springframework/expression/TypeLocator.java
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-
  jar/org/springframework/expression/MethodFilter.java
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-
  jar/org/springframework/expression/EvaluationException.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2002-2012 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*  
*      https://www.apache.org/licenses/LICENSE-2.0
*  
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-
  jar/org/springframework/expression/spel/InternalParseException.java
No license file was found, but licenses were detected in source scan.
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-jar/org/springframework/expression/ExpressionInvocationTargetException.java
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-jar/org/springframework/expression/common/TemplateParserContext.java
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-jar/org/springframework/expression/MethodResolver.java
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-jar/org/springframework/expression/MethodExecutor.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2014 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *   https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-jar/org/springframework/expression/spel/standard/SpelExpressionParser.java
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-jar/org/springframework/expression/spel/SpelCompilerMode.java
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-jar/org/springframework/expression/spel/CompilablePropertyAccessor.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2021 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *   https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and

Found in path(s):
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-jar/org/springframework/expression/BeanResolver.java
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-jar/org/springframework/expression/spel/ast/QualifiedIdentifier.java
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-jar/org/springframework/expression/spel/CodeFlow.java
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-jar/org/springframework/expression/spel/SpelMessage.java
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-jar/org/springframework/expression/ExpressionException.java
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-jar/org/springframework/expression/spel/support/ReflectiveConstructorResolver.java
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-jar/org/springframework/expression/spel/support/StandardTypeConverter.java
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-jar/org/springframework/expression/spel/ast/Operator.java
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-jar/org/springframework/expression/spel/ast/InlineMap.java
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-jar/org/springframework/expression/spel/SpelParserConfiguration.java
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-jar/org/springframework/expression/spel/standard/Token.java
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-jar/org/springframework/expression/spel/ast/BeanReference.java
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-jar/org/springframework/expression/spel/TypeComparator.java
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-jar/org/springframework/expression/spel/ast/TypeReference.java
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-jar/org/springframework/expression/spel/ast/ConstructorReference.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2019 the original author or authors.
 * 
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 *    https://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-
jar/org/springframework/expression/spel/ast/OperatorBetween.java
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-
jar/org/springframework/expression/spel/ast/Elvis.java
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-
jar/org/springframework/expression/spel/ast/OpPlus.java
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-
jar/org/springframework/expression/spel/ast/Projection.java
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-
jar/org/springframework/expression/spel/ast/OpMultiply.java
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-
jar/org/springframework/expression/spel/ast/OperatorNot.java
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-
jar/org/springframework/expression/spel/ast/StringLiteral.java
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-
jar/org/springframework/expression/spel/ast/Assign.java
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-
jar/org/springframework/expression/spel/ast/OpGT.java
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-
jar/org/springframework/expression/spel/ast/OpOr.java
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-
jar/org/springframework/expression/spel/ast/PropertyOrFieldReference.java
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-
jar/org/springframework/expression/spel/ast/NullLiteral.java
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-
jar/org/springframework/expression/spel/support/StandardTypeComparator.java
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-
jar/org/springframework/expression/spel/ast/Literal.java
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-
jar/org/springframework/expression/spel/ast/OpMinus.java
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-
jar/org/springframework/expression/spel/ast/SpelNodeImpl.java
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-
jar/org/springframework/expression/spel/ast/OpAnd.java
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-
jar/org/springframework/expression/spel/ast/OpDec.java
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-
jar/org/springframework/expression/spel/ast/CompoundExpression.java
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-
jar/org/springframework/expression/spel/ast/OperatorPower.java
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-
jar/org/springframework/expression/spel/ast/LongLiteral.java
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-
/*
 * Copyright 2002-2017 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-jar/org/springframework/expression/TypeConverter.java
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-jar/org/springframework/expression/common/ExpressionUtils.java
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-jar/org/springframework/expression/AccessException.java
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-jar/org/springframework/expression/spel/SpelParseException.java
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-jar/org/springframework/expression/OperatorOverloader.java
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-jar/org/springframework/expression/spel/support/ReflectiveConstructorExecutor.java
* /opt/cola/permits/1220178521_1635319533.02/0/spring-expression-5-3-12-sources-jar/org/springframework/expression/ParserContext.java

1.67 biweekly 0.5.0
1.67.1 Available under license:

Apache License
Version 2.0, January 2004
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and
(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License. You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NONINFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Copyright (c) 2013-2016, Michael Angstadt
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.68 netty-transport-rxtx 4.1.69.Final

1.68.1 Available under license:
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2013 The Netty Project
 */
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1235364932_1638382296.01/0/netty-transport-rxtx-4-1-69-final-sources-jar/io/netty/channel/rxtx/RxtxChannelOption.java
* /opt/cola/permits/1235364932_1638382296.01/0/netty-transport-rxtx-4-1-69-final-sources-jar/io/netty/channel/rxtx/RxtxChannelConfig.java
* /opt/cola/permits/1235364932_1638382296.01/0/netty-transport-rxtx-4-1-69-final-sources-jar/io/netty/channel/rxtx/RxtxChannel.java
* /opt/cola/permits/1235364932_1638382296.01/0/netty-transport-rxtx-4-1-69-final-sources-jar/io/netty/channel/rxtx/package-info.java
* /opt/cola/permits/1235364932_1638382296.01/0/netty-transport-rxtx-4-1-69-final-sources-jar/io/netty/channel/rxtx/DefaultRxtxChannelConfig.java
* /opt/cola/permits/1235364932_1638382296.01/0/netty-transport-rxtx-4-1-69-final-sources-jar/io/netty/channel/rxtx/RxtxDeviceAddress.java

No license file was found, but licenses were detected in source scan.

~ Copyright 2013 The Netty Project
~
~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~
~ https://www.apache.org/licenses/LICENSE-2.0
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.

Found in path(s):
* /opt/cola/permits/1235364932_1638382296.01/0/netty-transport-rxtx-4-1-69-final-sources-jar/META-INF/maven/io.netty/netty-transport-rxtx/pom.xml
1.69 stax-ex 1.7.8

1.70 commons-lang3 2.4

1.70.1 Available under license:

Apache Commons Lang
Copyright 2001-2008 The Apache Software Foundation

This product includes software developed by
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.
"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.
To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.71 java-driver-mapper-runtime 4.10.0
1.71.1 Available under license :

No license file was found, but licenses were detected in source scan.

Copyright DataStax, Inc.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Found in path(s):
* /opt/cola/permits/1145782256_1616451470.25/0/java-driver-mapper-runtime-4-10-0-sources-jar/META-INF/maven/com.datastax.oss/java-driver-mapper-runtime/pom.xml
No license file was found, but licenses were detected in source scan.

/*
 * Copyright DataStax, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1145782256_1616451470.25/0/java-driver-mapper-runtime-4-10-0-sources-jar/com/datastax/dse/driver/api/mapper/reactive/MappedReactiveResultSet.java
* /opt/cola/permits/1145782256_1616451470.25/0/java-driver-mapper-runtime-4-10-0-sources-jar/com/datastax/oss/driver/api/mapper/entity/naming/SetterStyle.java
* /opt/cola/permits/1145782256_1616451470.25/0/java-driver-mapper-runtime-4-10-0-sources-jar/com/datastax/oss/driver/api/mapper/annotations/NamingStrategy.java
* /opt/cola/permits/1145782256_1616451470.25/0/java-driver-mapper-runtime-4-10-0-sources-jar/com/datastax/driver/driver/internal/mapper/reactive/DefaultMappedReactiveResultSet.java
* /opt/cola/permits/1145782256_1616451470.25/0/java-driver-mapper-runtime-4-10-0-sources-jar/com/datastax/oss/driver/api/mapper/MapperContext.java
* /opt/cola/permits/1145782256_1616451470.25/0/java-driver-mapper-runtime-4-10-0-sources-jar/com/datastax/oss/driver/api/mapper/result/MapperResultProducer.java
* /opt/cola/permits/1145782256_1616451470.25/0/java-driver-mapper-runtime-4-10-0-sources-jar/com/datastax/oss/driver/api/mapper/annotations/QueryProvider.java
* /opt/cola/permits/1145782256_1616451470.25/0/java-driver-mapper-runtime-4-10-0-sources-jar/com/datastax/oss/driver/api/mapper/entity/naming/GetterStyle.java
* /opt/cola/permits/1145782256_1616451470.25/0/java-driver-mapper-runtime-4-10-0-sources-jar/com/datastax/oss/driver/api/mapper/entity/naming/NamingConvention.java
* /opt/cola/permits/1145782256_1616451470.25/0/java-driver-mapper-runtime-4-10-0-sources-jar/com/datastax/oss/driver/api/mapper/annotations/DaoTable.java
* /opt/cola/permits/1145782256_1616451470.25/0/java-driver-mapper-runtime-4-10-0-sources-jar/com/datastax/oss/driver/api/mapper/annotations/DaoKeyspace.java
* /opt/cola/permits/1145782256_1616451470.25/0/java-driver-mapper-runtime-4-10-0-sources-jar/com/datastax/oss/driver/api/mapper/annotations/Mapper.java
* /opt/cola/permits/1145782256_1616451470.25/0/java-driver-mapper-runtime-4-10-0-sources-jar/com/datastax/oss/driver/api/mapper/entity/saving/NullSavingStrategy.java
* /opt/cola/permits/1145782256_1616451470.25/0/java-driver-mapper-runtime-4-10-0-sources-jar/com/datastax/oss/driver/api/mapper/annotations/Entity.java
* /opt/cola/permits/1145782256_1616451470.25/0/java-driver-mapper-runtime-4-10-0-sources-
1.72 netty-resolver-dns 4.1.69.Final

1.72.1 Available under license:

No license file was found, but licenses were detected in source scan.

~ Copyright 2014 The Netty Project

~

~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~
~    https://www.apache.org/licenses/LICENSE-2.0
~

~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.

Found in path(s):

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015 The Netty Project
 */

* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1221869477_1635493931.72/0/netty-resolver-dns-4-1-69-final-sources-
ar/jar/io/netty/resolver/dns/RotationalDnsServerAddresses.java
* /opt/cola/permits/1221869477_1635493931.72/0/netty-resolver-dns-4-1-69-final-sources-
ar/jar/io/netty/resolver/dns/DefaultDnsServerAddresses.java
* /opt/cola/permits/1221869477_1635493931.72/0/netty-resolver-dns-4-1-69-final-sources-
ar/jar/io/netty/resolver/dns/DnsNameResolverBuilder.java
* /opt/cola/permits/1221869477_1635493931.72/0/netty-resolver-dns-4-1-69-final-sources-
ar/jar/io/netty/resolver/dns/DnsServerAddressStream.java
* /opt/cola/permits/1221869477_1635493931.72/0/netty-resolver-dns-4-1-69-final-sources-
ar/jar/io/netty/resolver/dns/DnsQueryContextManager.java
* /opt/cola/permits/1221869477_1635493931.72/0/netty-resolver-dns-4-1-69-final-sources-
ar/jar/io/netty/resolver/dns/DnsNameResolverException.java
* /opt/cola/permits/1221869477_1635493931.72/0/netty-resolver-dns-4-1-69-final-sources-
ar/jar/io/netty/resolver/dns/ShuffledDnsServerAddressStream.java
* /opt/cola/permits/1221869477_1635493931.72/0/netty-resolver-dns-4-1-69-final-sources-
ar/jar/io/netty/resolver/dns/SequentialDnsServerAddressStream.java
* /opt/cola/permits/1221869477_1635493931.72/0/netty-resolver-dns-4-1-69-final-sources-
ar/jar/io/netty/resolver/dns/SingletonDnsServerAddresses.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2019 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */
Found in path(s):
* /opt/cola/permits/1221869477_1635493931.72/0/netty-resolver-dns-4-1-69-final-sources-jar/io/netty/resolver/dns/PreferredAddressTypeComparator.java
* /opt/cola/permits/1221869477_1635493931.72/0/netty-resolver-dns-4-1-69-final-sources-jar/io/netty/resolver/dns/TcpDnsQueryContext.java

No license file was found, but licenses were detected in source scan.

# The Netty Project licenses this file to you under the Apache License,
# version 2.0 (the "License"); you may not use this file except in compliance
# with the License. You may obtain a copy of the License at:
# distributed under the License is distributed on an "AS IS" BASIS, WITHOUT

Found in path(s):

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2020 The Netty Project
 * 
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * 
 * https://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1221869477_1635493931.72/0/netty-resolver-dns-4-1-69-final-sources-jar/io/netty/resolver/dns/LoggingDnsQueryLifeCycleObserverFactory.java
* /opt/cola/permits/1221869477_1635493931.72/0/netty-resolver-dns-4-1-69-final-sources-jar/io/netty/resolver/dns/UnixResolverOptions.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2018 The Netty Project
 * 
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* https://www.apache.org/licenses/LICENSE-2.0
* Unauthorized use of this software is prohibited, unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1221869477_1635493931.72/0/netty-resolver-dns-4-1-69-final-sources-
  jar/io/netty/resolver/dns/AuthoritativeDnsServerCacheAdapter.java
* /opt/cola/permits/1221869477_1635493931.72/0/netty-resolver-dns-4-1-69-final-sources-
  jar/io/netty/resolver/dns/AuthoritativeDnsServerCache.java
* /opt/cola/permits/1221869477_1635493931.72/0/netty-resolver-dns-4-1-69-final-sources-
  jar/io/netty/resolver/dns/DirContextUtils.java
* /opt/cola/permits/1221869477_1635493931.72/0/netty-resolver-dns-4-1-69-final-sources-
  jar/io/netty/resolver/dns/DefaultDnsCnameCache.java
* /opt/cola/permits/1221869477_1635493931.72/0/netty-resolver-dns-4-1-69-final-sources-
  jar/io/netty/resolver/dns/DnsRecordResolveContext.java
* /opt/cola/permits/1221869477_1635493931.72/0/netty-resolver-dns-4-1-69-final-sources-
  jar/io/netty/resolver/dns/DnsAddressResolveContext.java
* /opt/cola/permits/1221869477_1635493931.72/0/netty-resolver-dns-4-1-69-final-sources-
  jar/io/netty/resolver/dns/Cache.java
* /opt/cola/permits/1221869477_1635493931.72/0/netty-resolver-dns-4-1-69-final-sources-
  jar/io/netty/resolver/dns/NameServerComparator.java
* /opt/cola/permits/1221869477_1635493931.72/0/netty-resolver-dns-4-1-69-final-sources-
  jar/io/netty/resolver/dns/DnsAddressDecoder.java
* /opt/cola/permits/1221869477_1635493931.72/0/netty-resolver-dns-4-1-69-final-sources-
  jar/io/netty/resolver/dns/NoopAuthoritativeDnsServerCache.java
* /opt/cola/permits/1221869477_1635493931.72/0/netty-resolver-dns-4-1-69-final-sources-
  jar/io/netty/resolver/dns/NoopDnsCnameCache.java
* /opt/cola/permits/1221869477_1635493931.72/0/netty-resolver-dns-4-1-69-final-sources-
  jar/io/netty/resolver/dns/DefaultAuthoritativeDnsServerCache.java
* /opt/cola/permits/1221869477_1635493931.72/0/netty-resolver-dns-4-1-69-final-sources-
  jar/io/netty/resolver/dns/DnsCnameCache.java

No license file was found, but licenses were detected in source scan.

/ *
* Copyright 2014 The Netty Project
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* https://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1221869477_1635493931.72/0/netty-resolver-dns-4-1-69-final-sources-
  jar/io/netty/resolver/dns/package-info.java
* /opt/cola/permits/1221869477_1635493931.72/0/netty-resolver-dns-4-1-69-final-sources-
  jar/io/netty/resolver/dns/DnsResolveContext.java
* /opt/cola/permits/1221869477_1635493931.72/0/netty-resolver-dns-4-1-69-final-sources-
  jar/io/netty/resolver/dns/DnsNameResolver.java
* /opt/cola/permits/1221869477_1635493931.72/0/netty-resolver-dns-4-1-69-final-sources-
  jar/io/netty/resolver/dns/DnsAddressResolverGroup.java
* /opt/cola/permits/1221869477_1635493931.72/0/netty-resolver-dns-4-1-69-final-sources-
  jar/io/netty/resolver/dns/DnsQueryContext.java
* /opt/cola/permits/1221869477_1635493931.72/0/netty-resolver-dns-4-1-69-final-sources-
  jar/io/netty/resolver/dns/DnsServerAddresses.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2016 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1221869477_1635493931.72/0/netty-resolver-dns-4-1-69-final-sources-
  jar/io/netty/resolver/dns/NoopDnsCache.java
* /opt/cola/permits/1221869477_1635493931.72/0/netty-resolver-dns-4-1-69-final-sources-
  jar/io/netty/resolver/dns/DnsCache.java
* /opt/cola/permits/1221869477_1635493931.72/0/netty-resolver-dns-4-1-69-final-sources-
  jar/io/netty/resolver/dns/RoundRobinDnsAddressResolverGroup.java
* /opt/cola/permits/1221869477_1635493931.72/0/netty-resolver-dns-4-1-69-final-sources-
  jar/io/netty/resolver/dns/DefaultDnsCache.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2017 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
  * /opt/cola/permits/1221869477_1635493931.72/0/netty-resolver-dns-4-1-69-final-sources-jar/io/netty/resolver/dns/InflightNameResolver.java
  * /opt/cola/permits/1221869477_1635493931.72/0/netty-resolver-dns-4-1-69-final-sources-jar/io/netty/resolver/dns/NoopDnsQueryLifecycleObserver.java
  * /opt/cola/permits/1221869477_1635493931.72/0/netty-resolver-dns-4-1-69-final-sources-jar/io/netty/resolver/dns/BiDnsQueryLifecycleObserver.java
  * /opt/cola/permits/1221869477_1635493931.72/0/netty-resolver-dns-4-1-69-final-sources-jar/io/netty/resolver/dns/MultiDnsServerAddressStreamProvider.java
  * /opt/cola/permits/1221869477_1635493931.72/0/netty-resolver-dns-4-1-69-final-sources-jar/io/netty/resolver/dns/SingletonDnsServerAddressStreamProvider.java
  * /opt/cola/permits/1221869477_1635493931.72/0/netty-resolver-dns-4-1-69-final-sources-jar/io/netty/resolver/dns/DefaultDnsServerAddressStreamProvider.java
  * /opt/cola/permits/1221869477_1635493931.72/0/netty-resolver-dns-4-1-69-final-sources-jar/io/netty/resolver/dns/UniSequentialDnsServerAddressStreamProvider.java
  * /opt/cola/permits/1221869477_1635493931.72/0/netty-resolver-dns-4-1-69-final-sources-jar/io/netty/resolver/dns/LoggingDnsQueryLifecycleObserver.java
  * /opt/cola/permits/1221869477_1635493931.72/0/netty-resolver-dns-4-1-69-final-sources-jar/io/netty/resolver/dns/UnixResolverDnsServerAddressStreamProvider.java
  * /opt/cola/permits/1221869477_1635493931.72/0/netty-resolver-dns-4-1-69-final-sources-jar/io/netty/resolver/dns/BiDnsQueryLifecycleObserverFactory.java
  * /opt/cola/permits/1221869477_1635493931.72/0/netty-resolver-dns-4-1-69-final-sources-jar/io/netty/resolver/dns/DnsCacheEntry.java
1.73 error_prone_annotations 2.5.1

1.73.1 Available under license:

No license file was found, but licenses were detected in source scan.

Copyright 2015 The Error Prone Authors.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Found in path(s):
* /opt/cola/permits/1146221464_1616477529.06/0/error-prone-annotations-2-5-1-jar/META-INF/maven/com.google.errorprone/error_prone_annotations/pom.xml

1.74 antlr-runtime 4.5

1.75 ucp 18.3.0.0

1.75.1 Available under license:

ISE uses the Oracle Database server under commercial license.

Cisco does not have a source code license thus it is impossible to upload any source code.

Mon Aug 17 10:11:49 PDT 2020 Martin Fujitani <martinf@cisco.com>
/* ====================================================================
* The Apache Software License, Version 1.1
*
* Copyright (c) 2000-2002 The Apache Software Foundation. All rights
* reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
*
* 1. Redistributions of source code must retain the above copyright
*    notice, this list of conditions and the following disclaimer.
*
* 2. Redistributions in binary form must reproduce the above copyright
*    notice, this list of conditions and the following disclaimer in
*    the documentation and/or other materials provided with the
*    distribution.
*
* 3. The end-user documentation included with the redistribution,
*    if any, must include the following acknowledgment:
*    "This product includes software developed by the
*    Apache Software Foundation (http://www.apache.org/)."
*    Alternately, this acknowledgment may appear in the software itself,
*    if and wherever such third-party acknowledgments normally appear.
*
* 4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
*    must not be used to endorse or promote products derived from this
*    software without prior written permission. For written
*    permission, please contact apache@apache.org.
*
* 5. Products derived from this software may not be called "Apache"
*    or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
*    name, without prior written permission of the Apache Software Foundation.
*
* THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.

*===================================================================
*
* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
*
* Portions of this software are based upon software originally written
* by Daniel F. Savarese. We appreciate his contributions.
*/

1.77 guava 30.1.1-jre
1.77.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.
"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or

Open Source Used in security_automation identity-guest-cs 434
for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason
END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.78 jetty-continuation 9.4.44.v20210927

1.78.1 Available under license:

This program and the accompanying materials are made available under the terms of the Eclipse Public License 2.0 which is available at http://www.eclipse.org/legal/epl-2.0, or the Apache Software License 2.0 which is available at https://www.apache.org/licenses/LICENSE-2.0.

Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS
"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and
b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor
disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and

b) its license agreement:
   i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
   ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;
   iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and
   iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and

b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is
intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must:

a) promptly notify the Commercial Contributor in writing of such claim, and
b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this
Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

====================================================================
Jetty Web Container
Copyright 1995-2018 Mort Bay Consulting Pty Ltd.
====================================================================

The Jetty Web Container is Copyright Mort Bay Consulting Pty Ltd
unless otherwise noted.

Jetty is dual licensed under both

* The Apache 2.0 License
  http://www.apache.org/licenses/LICENSE-2.0.html

  and

* The Eclipse Public 1.0 License

Jetty may be distributed under either license.

-----

Eclipse

The following artifacts are EPL.
* org.eclipse.jetty.orbit:org.eclipse.jdt.core

The following artifacts are EPL and ASL2.
* org.eclipse.jetty.orbit:javax.security.auth.message

The following artifacts are EPL and CDDL 1.0.
* org.eclipse.jetty.orbit:javax.mail.glassfish
Oracle

The following artifacts are CDDL + GPLv2 with classpath exception.
https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

* javax.servlet:javax.servlet-api
* javax.annotation:javax.annotation-api
* javax.transaction:javax.transaction-api
* javax.websocket:javax.websocket-api

Oracle OpenJDK

If ALPN is used to negotiate HTTP/2 connections, then the following artifacts may be included in the distribution or downloaded when ALPN module is selected.

* java.sun.security.ssl

These artifacts replace/modify OpenJDK classes. The modifications are hosted at github and both modified and original are under GPL v2 with classpath exceptions.
http://openjdk.java.net/legal/gplv2+ce.html

OW2

The following artifacts are licensed by the OW2 Foundation according to the terms of http://asm.ow2.org/license.html

org.ow2.asm:asm-commons
org.ow2.asm:asm

Apache

The following artifacts are ASL2 licensed.

org.apache.taglibs:taglibs-standard-spec
org.apache.taglibs:taglibs-standard-impl

MortBay
The following artifacts are ASL2 licensed. Based on selected classes from following Apache Tomcat jars, all ASL2 licensed.

org.mortbay.jasper:apache-jsp
org.apache.tomcat:tomcat-jasper
org.apache.tomcat:tomcat-juli
org.apache.tomcat:tomcat-jsp-api
org.apache.tomcat:tomcat-el-api
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-api
org.apache.tomcat:tomcat-util-scan
org.apache.tomcat:tomcat-util

org.mortbay.jasper:apache-el
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-el-api

------
Mortbay

The following artifacts are CDDL + GPLv2 with classpath exception.

https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

org.eclipse.jetty.toolchain:jetty-schemas

------
Assorted

The UnixCrypt.java code implements the one way cryptography used by Unix systems for simple password protection. Copyright 1996 Aki Yoshida, modified April 2001 by Iris Van den Broeke, Daniel Deville. Permission to use, copy, modify and distribute UnixCrypt for non-commercial or commercial purposes and without fee is granted provided that the copyright notice appears in all copies.

1.79 mail 1.4.4
1.79.1 Available under license:
COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.0

1. Definitions.

1.1. Contributor. means each individual or entity that creates or contributes to the creation of Modifications.
1.2. Contributor Version. means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. Covered Software. means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. Executable. means the Covered Software in any form other than Source Code.

1.5. Initial Developer. means the individual or entity that first makes Original Software available under this License.

1.6. Larger Work. means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. License. means this document.

1.8. Licensable. means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. Modifications. means the Source Code and Executable form of any of the following:

   A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

   B. Any new file that contains any part of the Original Software or previous Modification; or

   C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. Original Software. means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. Patent Claims. means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code. means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. You. (or .Your.) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, .You. includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, .control. means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims,
the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable
form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients' rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipients' rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.


You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.

Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer
includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.
When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as Participant.) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER
CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY’S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a .commercial item., as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of .commercial computer software. (as that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and .commercial computer software documentation. as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction’s conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys. fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)

The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-
law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc. 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.
Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not
responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM
"AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.

Copyright (C)

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and
you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

  signature of Ty Coon, 1 April 1989
  Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

"CLASSPATH" EXCEPTION TO THE GPL VERSION 2

Certain source files distributed by Sun Microsystems, Inc. are subject to the following clarification and special exception to the GPL Version 2, but only where Sun has expressly included in the particular source file's header the words

"Sun designates this particular file as subject to the "Classpath" exception as provided by Sun in the License file that accompanied this code."

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License Version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module.? An independent module is a module which is not derived from or based on this library.? If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so.? If you do not wish to do so, delete this exception statement from your version.

1.80 jsoup 1.11.3

1.80.1 Available under license:
The MIT License

Copyright (c) 2009-2018 Jonathan Hedley <jonathan@hedley.net>
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.81 jackson-dataformat-yaml 2.13.0
1.81.1 Available under license:

# Jackson JSON processor

Jackson is a high-performance, Free/Open Source JSON processing library. It was originally written by Tatu Saloranta (tatu.saloranta@iki.fi), and has been in development since 2007. It is currently developed by a community of developers, as well as supported commercially by FasterXML.com.

## Licensing

Jackson core and extension components may be licensed under different licenses. To find the details that apply to this artifact see the accompanying LICENSE file. For more information, including possible other licensing options, contact FasterXML.com (http://fasterxml.com).

## Credits

A list of contributors may be found from CREDITS file, which is included in some artifacts (usually source distributions); but is always available from the source code management (SCM) system project uses. This copy of Jackson JSON processor YAML module is licensed under the Apache (Software) License, version 2.0 ("the License"). See the License for details about distribution rights, and the specific rights regarding derivate works.

You may obtain a copy of the License at:
1.82 jackson-xc 1.9.2

1.82.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.83 classmate 1.5.1
1.83.1 Available under license:
Java ClassMate library was originally written by Tatu Saloranta (tatu.saloranta@iki.fi)

Other developers who have contributed code are:

* Brian Langel
This copy of Java ClassMate library is licensed under Apache (Software) License, version 2.0 ("the License"). See the License for details about distribution rights, and the specific rights regarding derivate works.

You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

1.84 json-smart 2.3
1.84.1 Available under license:
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2011 JSON-SMART authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at
 */
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*
Found in path(s):
* /opt/cola/permits/1177952445_1625583906.57/0/json-smart-2-3-sources-4-jar/net/minidev/json/JSONObject.java
* /opt/cola/permits/1177952445_1625583906.57/0/json-smart-2-3-sources-4-jar/net/minidev/json/JSONNavi.java
* /opt/cola/permits/1177952445_1625583906.57/0/json-smart-2-3-sources-4-jar/net/minidev/json/JSONArray.java
* /opt/cola/permits/1177952445_1625583906.57/0/json-smart-2-3-sources-4-jar/net/minidev/json/JSONStreamAwareEx.java
* /opt/cola/permits/1177952445_1625583906.57/0/json-smart-2-3-sources-4-jar/net/minidev/json/JSONAwareEx.java
* /opt/cola/permits/1177952445_1625583906.57/0/json-smart-2-3-sources-4-jar/net/minidev/json/FakeMapper.java
* /opt/cola/permits/1177952445_1625583906.57/0/json-smart-2-3-sources-4-jar/net/minidev/json/JSONParserString.java
* /opt/cola/permits/1177952445_1625583906.57/0/json-smart-2-3-sources-4-jar/net/minidev/json/BeansMapper.java
* /opt/cola/permits/1177952445_1625583906.57/0/json-smart-2-3-sources-4-jar/net/minidev/json/JSONParserBase.java
* /opt/cola/permits/1177952445_1625583906.57/0/json-smart-2-3-sources-4-jar/net/minidev/json/JSONParserReader.java
* /opt/cola/permits/1177952445_1625583906.57/0/json-smart-2-3-sources-4-jar/net/minidev/json/JSONStreamAware.java
* /opt/cola/permits/1177952445_1625583906.57/0/json-smart-2-3-sources-4-jar/net/minidev/json/DefaultMapperOrdered.java
* /opt/cola/permits/1177952445_1625583906.57/0/json-smart-2-3-sources-4-jar/net/minidev/json/JSONParserMemory.java
* /opt/cola/permits/1177952445_1625583906.57/0/json-smart-2-3-sources-4-jar/net/minidev/json/ArraysMapper.java
* /opt/cola/permits/1177952445_1625583906.57/0/json-smart-2-3-sources-4-jar/net/minidev/json/CollectionMapper.java
* /opt/cola/permits/1177952445_1625583906.57/0/json-smart-2-3-sources-4-jar/net/minidev/json/CompessorMapper.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2011-2014 JSON-SMART authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1177952445_1625583906.57/0/json-smart-2-3-sources-4-jar/net/minidev/json/parser/DefaultMapper.java

1.85 jmespath-java 1.11.868

1.85.1 Available under license:
No license file was found, but licenses were detected in source scan.

<name>Apache License, Version 2.0</name>

Found in path(s):
* /opt/cola/permits/1265673601_1643932655.02/0/jmespath-java-1-11-868-sources-jar/META-INF/maven/com.amazonaws/jmespath-java/pom.xml

No license file was found, but licenses were detected in source scan.

/
* Copyright 2010-2020 Amazon.com, Inc. or its affiliates. All Rights Reserved.
* Licensed under the Apache License, Version 2.0 (the "License").
* You may not use this file except in compliance with the License.
* A copy of the License is located at
* http://aws.amazon.com/apache2.0
* or in the "license" file accompanying this file. This file is distributed
* on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express or implied. See the License for the specific language governing
* permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1265673601_1643932655.02/0/jmespath-java-1-11-868-sources.jar/com/amazonaws/jmespath/Comparator.java
* /opt/cola/permits/1265673601_1643932655.02/0/jmespath-java-1-11-868-sources.jar/com/amazonaws/jmespath/OpGreaterThanOrEqualTo.java
* /opt/cola/permits/1265673601_1643932655.02/0/jmespath-java-1-11-868-sources.jar/com/amazonaws/jmespath/JmesPathVisitor.java
* /opt/cola/permits/1265673601_1643932655.02/0/jmespath-java-1-11-868-sources.jar/com/amazonaws/jmespath/JmesPathFilter.java
* /opt/cola/permits/1265673601_1643932655.02/0/jmespath-java-1-11-868-sources.jar/com/amazonaws/jmespath/JmesPathContainsFunction.java
* /opt/cola/permits/1265673601_1643932655.02/0/jmespath-java-1-11-868-sources.jar/com/amazonaws/jmespath/JmesPathNotExpression.java
* /opt/cola/permits/1265673601_1643932655.02/0/jmespath-java-1-11-868-sources.jar/com/amazonaws/jmespath/JmesPathField.java
* /opt/cola/permits/1265673601_1643932655.02/0/jmespath-java-1-11-868-sources.jar/com/amazonaws/jmespath/OpGreaterThan.java
* /opt/cola/permits/1265673601_1643932655.02/0/jmespath-java-1-11-868-sources.jar/com/amazonaws/jmespath/OpLessThanOrEqualTo.java
* /opt/cola/permits/1265673601_1643932655.02/0/jmespath-java-1-11-868-sources.jar/com/amazonaws/jmespath/OpGreaterThan.java
* /opt/cola/permits/1265673601_1643932655.02/0/jmespath-java-1-11-868-sources.jar/com/amazonaws/jmespath/OpLessThanOrEqualTo.java
* /opt/cola/permits/1265673601_1643932655.02/0/jmespath-java-1-11-868-sources.jar/com/amazonaws/jmespath/ObjectMapperSingleton.java
* /opt/cola/permits/1265673601_1643932655.02/0/jmespath-java-1-11-868-sources.jar/com/amazonaws/jmespath/JmesPathProjection.java
* /opt/cola/permits/1265673601_1643932655.02/0/jmespath-java-1-11-868-sources.jar/com/amazonaws/jmespath/JmesPathAndExpression.java
* /opt/cola/permits/1265673601_1643932655.02/0/jmespath-java-1-11-868-sources.jar/com/amazonaws/jmespath/JmesPathExpression.java
* /opt/cola/permits/1265673601_1643932655.02/0/jmespath-java-1-11-868-sources.jar/com/amazonaws/jmespath/JmesPathLiteral.java
* /opt/cola/permits/1265673601_1643932655.02/0/jmespath-java-1-11-868-sources-
1.86 metrics-httpclient 4.1.26

1.86.1 Available under license:

Apache License

Version 2.0, January 2004

http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.
"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
You must give any other recipients of the Work or Derivative Works a copy of this License; and
You must cause any modified files to carry prominent notices stating that You changed the files; and
You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark,
and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of
the Derivative Works; and
If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute
must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices
that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text
file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the
Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices
normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the
License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an
addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license
terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works
as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions
stated in this License.
5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for
inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any
additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of
any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product
names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and
reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work
(and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR
CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or
conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR
PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and
assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or
otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing,
shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or
consequential damages of any character arising as a result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or
any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of
such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may
choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or
rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf
and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend,
and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: HOW TO APPLY THE APACHE LICENSE TO YOUR WORK
To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[ ]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.87 jain-sip-ri 1.2.148.3

1.87.1 Available under license:

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Sipp default 'uac' scenario.

/*
 * Conditions Of Use
This software was developed by employees of the National Institute of Standards and Technology (NIST), an agency of the Federal Government. Pursuant to title 15 United States Code Section 105, works of NIST employees are not subject to copyright protection in the United States and are considered to be in the public domain. As a result, a formal license is not needed to use the software.

This software is provided by NIST as a service and is expressly provided "AS IS." NIST makes no warranty of any kind, express, implied or statutory, including, without limitation, the implied warranty of merchantability, fitness for a particular purpose, non-infringement and data accuracy. NIST does not warrant or make any representations regarding the use of the software or the results thereof, including but not limited to the correctness, accuracy, reliability or usefulness of the software.

Permission to use this software is contingent upon your acceptance of the terms of this agreement.

GNU GENERAL PUBLIC LICENSE

Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Lesser General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it...
in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of
running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the
Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not
compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the
integrity of the free software distribution system, which is
implemented by public license practices. Many people have made
generous contributions to the wide range of software distributed
through that system in reliance on consistent application of that
system; it is up to the author/donor to decide if he or she is willing
to distribute software through any other system and a licensee cannot
impose that choice.

This section is intended to make thoroughly clear what is believed to
be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in
certain countries either by patents or by copyrighted interfaces, the
original copyright holder who places the Program under this License
may add an explicit geographical distribution limitation excluding
those countries, so that distribution is permitted only in or among
countries not thus excluded. In such case, this License incorporates
the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions
of the General Public License from time to time. Such new versions will
be similar in spirit to the present version, but may differ in detail to
address new problems or concerns.

Each version is given a distinguishing version number. If the Program
specifies a version number of this License which applies to it and "any
later version", you have the option of following the terms and conditions
either of that version or of any later version published by the Free
Software Foundation. If the Program does not specify a version number of
this License, you may choose any version ever published by the Free Software
Foundation.

10. If you wish to incorporate parts of the Program into other free
programs whose distribution conditions are different, write to the author
to ask for permission. For software which is copyrighted by the Free
Software Foundation, write to the Free Software Foundation; we sometimes
make exceptions for this. Our decision will be guided by the two goals
of preserving the free status of all derivatives of our free software and
of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY
FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN
OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES
PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED
OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS
TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE
PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING,
REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING
WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR
REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES,
INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING
OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED
TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY
YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER
PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE
POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest
possible use to the public, the best way to achieve this is to make it
free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest
to attach them to the start of each source file to most effectively
convey the exclusion of warranty; and each file should have at least
the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License along
with this program; if not, write to the Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this
when it starts in an interactive mode:
Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type 'show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type 'show c' for details.

The hypothetical commands 'show w' and 'show c' should show the appropriate
parts of the General Public License. Of course, the commands you use may
be called something other than 'show w' and 'show c'; they could even be
mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the program, if
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program
'Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into
proprietary programs. If your program is a subroutine library, you may
consider it more useful to permit linking proprietary applications with the
library. If this is what you want to do, use the GNU Lesser General
Public License instead of this License.

1.88 protobuf-java 2.2.0

1.89 jetty-util 9.4.44.v20210927
1.89.1 Available under license :

This program and the accompanying materials are made available under the
terms of the Eclipse Public License 2.0 which is available at
http://www.eclipse.org/legal/epl-2.0, or the Apache Software License
2.0 which is available at https://www.apache.org/licenses/LICENSE-2.0.

Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC
LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM
CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.
1. DEFINITIONS

"Contribution” means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and
b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution ‘originates’ from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor” means any person or entity that distributes the Program.

"Licensed Patents” mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program” means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any
Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and
b) its license agreement:
   i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
   ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;
   iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and
   iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and
b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION
Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor (“Commercial Contributor”) hereby agrees to defend and indemnify every other Contributor (“Indemnified Contributor”) against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must:
a) promptly notify the Commercial Contributor in writing of such claim, and
b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However,Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.
This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Jetty Web Container
Copyright 1995-2018 Mort Bay Consulting Pty Ltd.

The Jetty Web Container is Copyright Mort Bay Consulting Pty Ltd unless otherwise noted.

Jetty is dual licensed under both

* The Apache 2.0 License
  http://www.apache.org/licenses/LICENSE-2.0.html

and

* The Eclipse Public 1.0 License

Jetty may be distributed under either license.

------
Eclipse

The following artifacts are EPL.
* org.eclipse.jetty.orbit:org.eclipse.jdt.core

The following artifacts are EPL and ASL2.
* org.eclipse.jetty.orbit:javax.security.auth.message
* org.eclipse.jetty.orbit:javax.mail.glassfish

------

Oracle

The following artifacts are CDDL + GPLv2 with classpath exception.
https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

* javax.servlet:javax.servlet-api
* javax.annotation:javax.annotation-api
* javax.transaction:javax.transaction-api
* javax.websocket:javax.websocket-api

------

Oracle OpenJDK

If ALPN is used to negotiate HTTP/2 connections, then the following artifacts may be included in the distribution or downloaded when ALPN module is selected.

* java.sun.security.ssl

These artifacts replace/modify OpenJDK classes. The modifications are hosted at github and both modified and original are under GPL v2 with classpath exceptions.
http://openjdk.java.net/legal/gplv2+ce.html

------

OW2

The following artifacts are licensed by the OW2 Foundation according to the terms of http://asm.ow2.org/license.html

org.ow2.asm:asm-commons
org.ow2.asm:asm

------

Apache

The following artifacts are ASL2 licensed.

org.apache.taglibs:taglibs-standard-spec
org.apache.taglibs:taglibs-standard-impl
MortBay

The following artifacts are ASL2 licensed. Based on selected classes from following Apache Tomcat jars, all ASL2 licensed.

org.mortbay.jasper:apache-jsp
org.apache.tomcat:tomcat-jasper
org.apache.tomcat:tomcat-juli
org.apache.tomcat:tomcat-jsp-api
org.apache.tomcat:tomcat-el-api
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-api
org.apache.tomcat:tomcat-util-scan
org.apache.tomcat:tomcat-util

org.mortbay.jasper:apache-el
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-el-api

MortBay

The following artifacts are CDDL + GPLv2 with classpath exception.

https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

org.eclipse.jetty.toolchain:jetty-schemas

Assorted

The UnixCrypt.java code implements the one way cryptography used by Unix systems for simple password protection. Copyright 1996 Aki Yoshida, modified April 2001 by Iris Van den Broeke, Daniel Deville. Permission to use, copy, modify and distribute UnixCrypt for non-commercial or commercial purposes and without fee is granted provided that the copyright notice appears in all copies.
1.91 jaxb-api 2.3.0

1.91.1 Available under license:

COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.1

1. Definitions.

1.1. "Contributor" means each individual or entity that creates or contributes to the creation of Modifications.

1.2. "Contributor Version" means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. "Covered Software" means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. "Executable" means the Covered Software in any form other than Source Code.

1.5. "Initial Developer" means the individual or entity that first makes Original Software available under this License.

1.6. "Larger Work" means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. "License" means this document.

1.8. "Licensable" means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. "Modifications" means the Source Code and Executable form of any of the following:

A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

B. Any new file that contains any part of the Original Software or previous Modification; or

C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. "Original Software" means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. "Patent Claims" means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. "Source Code" means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. "You" (or "Your") means an individual or a legal entity exercising rights under, and complying with all of
the terms of, this License. For legal entities, "You" includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the
Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients' rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.

Oracle is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.
6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as "Participant") alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. If You assert a patent infringement claim against Participant alleging that the Participant Software directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

6.4. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY’S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a "commercial item," as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of "commercial computer software" (as that term is defined at 48 C.F.R. ? 252.227-7014(a)(1)) and "commercial computer software documentation" as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License
is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

--------
NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)
The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc. 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to
make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish),
that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new
free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to
surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the
software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all
the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show
them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you
legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no
warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to
know that what they have is not the original, so that any problems introduced by others will not reflect on the
original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that
redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To
prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying
it may be distributed under the terms of this General Public License. The "Program", below, refers to any such
program or work, and a "work based on the Program" means either the Program or any derivative work under
copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications
and/or translated into another language. (Hereinafter, translation is included without limitation in the term
"modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its
scope. The act of running the Program is not restricted, and the output from the Program is covered only if its
contents constitute a work based on the Program (independent of having been made by running the Program).
Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium,
provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and
disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and
give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty
protection in exchange for a fee.
2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Therefore, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)
The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted
interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.
To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.

Copyright (C)

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

"CLASSPATH" EXCEPTION TO THE GPL VERSION 2

Certain source files distributed by Oracle are subject to the following clarification and special exception to the GPL
Version 2, but only where Oracle has expressly included in the particular source file's header the words "Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the License file that accompanied this code."

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License Version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

1.92 libffi 1.2

1.92.1 Available under license:

No license file was found, but licenses were detected in source scan.

Manifest-Version: 1.0
Archiver-Version: Plexus Archiver
Created-By: Apache Maven Bundle Plugin
Automatic-Module-Name: org.jnrproject.jffi
Bnd-LastModified: 1599864364736
Build-Jdk: 11.0.4
Built-By: headius
Bundle-ClassPath: .,jni
Bundle-Description: Java Foreign Function Interface - Native Libraries
Bundle-License: http://www.apache.org/licenses/LICENSE-2.0.txt
Bundle-ManifestVersion: 2
Bundle-Name: jffi
Bundle-SymbolicName: com.github.jnr.jffi.native
Bundle-Version: 1.3.0
Fragment-Host: com.github.jnr.jffi
Require-Bundle: com.github.jnr.jffi
Tool: Bnd-2.1.0.20130426-122213

Found in path(s):
* /opt/cola/permits/1242030553_1645092167.83/0/jffi-1-3-0-native-jar/META-INF/MANIFEST.MF

1.93 dom4j 1.6.1
1.93.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
File: ConcurrentReaderHashMap

Written by Doug Lea. Adapted and released, under explicit
permission, from JDK1.2 HashMap.java and Hashtable.java which
carries the following copyright:

* Copyright 1997 by Sun Microsystems, Inc.,
* 901 San Antonio Road, Palo Alto, California, 94303, U.S.A.
* All rights reserved.
*
* This software is the confidential and proprietary information
* of Sun Microsystems, Inc. ("Confidential Information"). You
* shall not disclose such Confidential Information and shall use
* it only in accordance with the terms of the license agreement
* you entered into with Sun.

History:

<table>
<thead>
<tr>
<th>Date</th>
<th>Who</th>
<th>What</th>
</tr>
</thead>
<tbody>
<tr>
<td>28oct1999</td>
<td>dl</td>
<td>Created</td>
</tr>
<tr>
<td>14dec1999</td>
<td>dl</td>
<td>jmm snapshot</td>
</tr>
<tr>
<td>19apr2000</td>
<td>dl</td>
<td>use barrierLock</td>
</tr>
<tr>
<td>12jan2001</td>
<td>dl</td>
<td>public release</td>
</tr>
<tr>
<td>17nov2001</td>
<td>dl</td>
<td>Minor tunings</td>
</tr>
<tr>
<td>20may2002</td>
<td>dl</td>
<td>BarrierLock can now be serialized.</td>
</tr>
<tr>
<td>09dec2002</td>
<td>dl</td>
<td>Fix interference checks.</td>
</tr>
</tbody>
</table>

*/

Found in path(s):
* /opt/cola/permits/1150916825_1617719489.56/0/dom4j-1-6-1-sources-4-
  jar/org/dom4j/tree/ConcurrentReaderHashMap.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2001-2005 (C) MetaStuff, Ltd. All Rights Reserved.
* This software is open source.
* See the bottom of this file for the licence.
*/

/*
* Redistribution and use of this software and associated documentation
* ("Software"), with or without modification, are permitted provided that the
* following conditions are met:
*
* 1. Redistributions of source code must retain copyright statements and
* notices. Redistributions must also contain a copy of this document.
*  
* 2. Redistributions in binary form must reproduce the above copyright notice,
* this list of conditions and the following disclaimer in the documentation
* and/or other materials provided with the distribution.
*  
* 3. The name "DOM4J" must not be used to endorse or promote products derived
* from this Software without prior written permission of MetaStuff, Ltd. For
* written permission, please contact dom4j-info@metastuff.com.
*  
* 4. Products derived from this Software may not be called "DOM4J" nor may
* "DOM4J" appear in their names without prior written permission of MetaStuff,
* Ltd. DOM4J is a registered trademark of MetaStuff, Ltd.
*  
* 5. Due credit should be given to the DOM4J Project - http://www.dom4j.org
*  
* THIS SOFTWARE IS PROVIDED BY METASTUFF, LTD. AND CONTRIBUTORS "AS IS" AND
* ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL METASTUFF, LTD. OR ITS CONTRIBUTORS BE
* LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
* CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
* SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
* INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
* CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE
* POSSIBILITY OF SUCH DAMAGE.
*  
* Copyright 2001-2005 (C) MetaStuff, Ltd. All Rights Reserved.
*/

Found in path(s):
* /opt/cola/permits/1150916825_1617719489.56/0/dom4j-1-6-1-sources-4-jar/org/dom4j/InvalidXPathException.java
* /opt/cola/permits/1150916825_1617719489.56/0/dom4j-1-6-1-sources-4-jar/org/dom4j/ProcessingInstruction.java
* /opt/cola/permits/1150916825_1617719489.56/0/dom4j-1-6-1-sources-4-jar/org/dom4j/swing/LeafTreeNode.java
* /opt/cola/permits/1150916825_1617719489.56/0/dom4j-1-6-1-sources-4-jar/org/dom4j/XPath.java
* /opt/cola/permits/1150916825_1617719489.56/0/dom4j-1-6-1-sources-4-jar/org/dom4j/util/IndexedElement.java
* /opt/cola/permits/1150916825_1617719489.56/0/dom4j-1-6-1-sources-4-jar/org/dom4j/tree/DefaultAttribute.java
* /opt/cola/permits/1150916825_1617719489.56/0/dom4j-1-6-1-sources-4-jar/org/dom4j/Branch.java
* /opt/cola/permits/1150916825_1617719489.56/0/dom4j-1-6-1-sources-4-jar/org/dom4j/swing/XMLTableColumnDefinition.java
* /opt/cola/permits/1150916825_1617719489.56/0/dom4j-1-6-1-sources-4-jar/org/dom4j/util/AttributeHelper.java
* /opt/cola/permits/1150916825_1617719489.56/0/dom4j-1-6-1-sources-4-jar/org/dom4j/util/NodeComparator.java
* /opt/cola/permits/1150916825_1617719489.56/0/dom4j-1-6-1-sources-4-jar/org/dom4j/tree/SingleIterator.java
* /opt/cola/permits/1150916825_1617719489.56/0/dom4j-1-6-1-sources-4-jar/org/dom4j/io/XPPReader.java
* /opt/cola/permits/1150916825_1617719489.56/0/dom4j-1-6-1-sources-4-jar/org/dom4j/Node.java
* /opt/cola/permits/1150916825_1617719489.56/0/dom4j-1-6-1-sources-4-jar/org/dom4j/rule/NullAction.java
1.94 jackson-datatype-guava 2.13.0
1.94.1 Available under license:

This copy of Jackson JSON processor `jackson-datatype-guava` module is licensed under the Apache (Software) License, version 2.0 ("the License"). See the License for details about distribution rights, and the specific rights regarding derivative works.

You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

1.95 guava 24.1.1-jre

1.96 jackson-annotations 2.13.0
1.96.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION
1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.97 jetty-io 9.4.44.v20210927

1.97.1 Available under license:
This program and the accompanying materials are made available under the terms of the Eclipse Public License 2.0 which is available at http://www.eclipse.org/legal/epl-2.0, or the Apache Software License 2.0 which is available at https://www.apache.org/licenses/LICENSE-2.0.

Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution” means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and
b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor” means any person or entity that distributes the Program.

"Licensed Patents” mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program” means the Contributions distributed in accordance with this Agreement.
"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS
   a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.
   b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.
   c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.
   d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

   A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

   a) it complies with the terms and conditions of this Agreement; and
   b) its license agreement:
      i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
      ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and
consequential damages, such as lost profits;
iii) states that any provisions which differ from this Agreement are
offered by that Contributor alone and not by any other party; and
iv) states that source code for the Program is available from such
Contributor, and informs licensees how to obtain it in a reasonable
manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and
b) a copy of this Agreement must be included with each copy of the Program.
Contributors may not remove or alter any copyright notices contained
within the Program.

Each Contributor must identify itself as the originator of its Contribution,
if
any, in a manner that reasonably allows subsequent Recipients to identify the
originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with
respect to end users, business partners and the like. While this license is
intended to facilitate the commercial use of the Program, the Contributor who
includes the Program in a commercial product offering should do so in a manner
which does not create potential liability for other Contributors. Therefore,
if a Contributor includes the Program in a commercial product offering, such
Contributor ("Commercial Contributor") hereby agrees to defend and indemnify
every other Contributor ("Indemnified Contributor") against any losses,
damages and costs (collectively "Losses") arising from claims, lawsuits and
other legal actions brought by a third party against the Indemnified
Contributor to the extent caused by the acts or omissions of such Commercial
Contributor in connection with its distribution of the Program in a commercial
product offering. The obligations in this section do not apply to any claims
or Losses relating to any actual or alleged intellectual property
infringement. In order to qualify, an Indemnified Contributor must:
a) promptly notify the Commercial Contributor in writing of such claim, and
b) allow the Commercial Contributor to control, and cooperate with the
Commercial Contributor in, the defense and any related settlement
negotiations. The Indemnified Contributor may participate in any such claim at
its own expense.

For example, a Contributor might include the Program in a commercial product
offering, Product X. That Contributor is then a Commercial Contributor. If
that Commercial Contributor then makes performance claims, or offers
warranties related to Product X, those performance claims and warranties are
such Commercial Contributor's responsibility alone. Under this section, the
Commercial Contributor would have to defend claims against the other
Contributors related to those performance claims and warranties, and if a
court requires any other Contributor to pay any damages as a result, the
Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN
"AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR
IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE,
NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each
Recipient is solely responsible for determining the appropriateness of using
and distributing the Program and assumes all risks associated with its
exercise of rights under this Agreement, including but not limited to the
risks and costs of program errors, compliance with applicable laws, damage to
or loss of data, programs or equipment, and unavailability or interruption of
operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY
CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION
LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE
EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY
OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under
applicable law, it shall not affect the validity or enforceability of the
remainder of the terms of this Agreement, and without further action by the
parties hereto, such provision shall be reformed to the minimum extent
necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Program itself
(excluding combinations of the Program with other software or hardware)
infringes such Recipient's patent(s), then such Recipient's rights granted
under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to
comply with any of the material terms or conditions of this Agreement and does
not cure such failure in a reasonable period of time after becoming aware of
such noncompliance. If all Recipient's rights under this Agreement terminate,
Recipient agrees to cease use and distribution of the Program as soon as
reasonably practicable. However, Recipient's obligations under this Agreement
and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

=====================================================================
Jetty Web Container
Copyright 1995-2018 Mort Bay Consulting Pty Ltd.
=====================================================================

The Jetty Web Container is Copyright Mort Bay Consulting Pty Ltd unless otherwise noted.

Jetty is dual licensed under both
* The Apache 2.0 License
http://www.apache.org/licenses/LICENSE-2.0.html

and

* The Eclipse Public 1.0 License

Jetty may be distributed under either license.

------

Eclipse

The following artifacts are EPL.
* org.eclipse.jetty.orbit:org.eclipse.jdt.core

The following artifacts are EPL and ASL2.
* org.eclipse.jetty.orbit:javax.security.auth.message

The following artifacts are EPL and CDDL 1.0.
* org.eclipse.jetty.orbit:javax.mail.glassfish

------

Oracle

The following artifacts are CDDL + GPLv2 with classpath exception.
https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

* javax.servlet:javax.servlet-api
* javax.annotation:javax.annotation-api
* javax.transaction:javax.transaction-api
* javax.websocket:javax.websocket-api

------

Oracle OpenJDK

If ALPN is used to negotiate HTTP/2 connections, then the following artifacts may be included in the distribution or downloaded when ALPN module is selected.

* java.sun.security.ssl

These artifacts replace/modify OpenJDK classes. The modifications are hosted at github and both modified and original are under GPL v2 with classpath exceptions.
http://openjdk.java.net/legal/gplv2+ce.html
-------

OW2

The following artifacts are licensed by the OW2 Foundation according to the terms of http://asm.ow2.org/license.html

org.ow2.asm:asm-commons
org.ow2.asm:asm

-------

Apache

The following artifacts are ASL2 licensed.

org.apache.taglibs:taglibs-standard-spec
org.apache.taglibs:taglibs-standard-impl

-------

MortBay

The following artifacts are ASL2 licensed. Based on selected classes from following Apache Tomcat jars, all ASL2 licensed.

org.mortbay.jasper:apache-jsp
org.apache.tomcat:tomcat-jasper
org.apache.tomcat:tomcat-juli
org.apache.tomcat:tomcat-jsp-api
org.apache.tomcat:tomcat-el-api
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-api
org.apache.tomcat:tomcat-util-scan
org.apache.tomcat:tomcat-util

org.mortbay.jasper:apache-el
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-el-api

-------

Mortbay

The following artifacts are CDDL + GPLv2 with classpath exception.

https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html
The UnixCrypt.java code implements the one way cryptography used by Unix systems for simple password protection. Copyright 1996 Aki Yoshida, modified April 2001 by Iris Van den Broeke, Daniel Deville.
Permission to use, copy, modify and distribute UnixCrypt for non-commercial or commercial purposes and without fee is granted provided that the copyright notice appears in all copies.

1.98 framework-supports 2.1.25-SNAPSHOT
1.98.1 Available under license :
No license file was found, but licenses were detected in source scan.

# confidentiality *AND* message authenticity. (NIST refers to such cipher

Found in path(s):
* /opt/cola/permits/1257811120_1646347962.41/0/framework-supports-2-1-25-snapshot-jar/esapi/ESAPI.properties

1.99 oauth2-oidc-sdk 8.36.2
1.99.1 Available under license :
No license file was found, but licenses were detected in source scan.

<name>Apache License, version 2.0</name>

Found in path(s):
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources-jar/META-INF/maven/com.nimbusds/oauth2-oidc-sdk/pom.xml
No license file was found, but licenses were detected in source scan.

/ *
* oauth2-oidc-sdk
*
* Copyright 2012-2016, Connect2id Ltd and contributors.
* *
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use
* this file except in compliance with the License. You may obtain a copy of the
* License at
* *
* http://www.apache.org/licenses/LICENSE-2.0
OpenID Connect client information. Encapsulates the registration and metadata details of an OpenID Connect client:

- The client identifier.
- The client OpenID Connect metadata.
- The optional client secret for a confidential client.
- The optional registration URI and access token if dynamic client registration is permitted.

Related specifications:

- OpenID Connect Dynamic Client Registration 1.0.
- OAuth 2.0 Dynamic Client Registration Protocol (RFC 7591), section 3.2.1.
- OAuth 2.0 Dynamic Client Registration Management Protocol (RFC 7592), section 3.

Found in path(s):

```
/opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources-jar/com/nimbusds/openid/connect/sdk/rp/OIDCClientInformation.java
```

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright 2012-2016, Connect2id Ltd and contributors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the
```
* specific language governing permissions and limitations under the License.
*/
/**
* Confidential. Clients capable of maintaining the confidentiality of
* their credentials (e.g., client implemented on a secure server with
* restricted access to the client credentials), or capable of secure
* client authentication using other means.
*/
/**
* Public. Clients incapable of maintaining the confidentiality of their
* credentials (e.g., clients executing on the device used by the
* resource owner, such as an installed native application or a web
* browser-based application), and incapable of secure client
* authentication via any other means.
*/

Found in path(s):
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources-
  jar/com/nimbusds/oauth2/sdk/client/ClientType.java
No license file was found, but licenses were detected in source scan.

/*
* oauth2-oidc-sdk
*
* Copyright 2012-2020, Connect2id Ltd and contributors.
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use
* this file except in compliance with the License. You may obtain a copy of the
* License at
*
*     http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed
* under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR
* CONDITIONS OF ANY KIND, either express or implied. See the License for the
* specific language governing permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources-
  jar/com/nimbusds/openid/connect/sdk/federation/policy/operations/PolicyOperationCombinationValidator.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources-
  jar/com/nimbusds/openid/connect/sdk/federation/entities/EntityRole.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources-
  jar/com/nimbusds/openid/connect/sdk/federation/trust/DefaultTrustChainRetriever.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources-
  jar/com/nimbusds/openid/connect/sdk/rp/statement/package-info.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources-
jar/com/nimbusds/openid/connect/sdk/federation/policy/operations/PolicyOperationFactory.java
* /opt/cola/permissions/1199843396_1630602842.72/oauth2-oidc-sdk-8-36-2-sources-
jar/com/nimbusds/openid/connect/sdk/federation/trust/constraints/ExactMatchEntityIDConstraint.java
* /opt/cola/permissions/1199843396_1630602842.72/oauth2-oidc-sdk-8-36-2-sources-
jar/com/nimbusds/openid/connect/sdk/OIDCClaimsRequest.java
* /opt/cola/permissions/1199843396_1630602842.72/oauth2-oidc-sdk-8-36-2-sources-
jar/com/nimbusds/openid/connect/sdk/federation/entities/package-info.java
* /opt/cola/permissions/1199843396_1630602842.72/oauth2-oidc-sdk-8-36-2-sources-
jar/com/nimbusds/openid/connect/sdk/federation/trust/marks/package-info.java
* /opt/cola/permissions/1199843396_1630602842.72/oauth2-oidc-sdk-8-36-2-sources-
jar/com/nimbusds/openid/connect/sdk/federation/api/EntityListingSpec.java
* /opt/cola/permissions/1199843396_1630602842.72/oauth2-oidc-sdk-8-36-2-sources-
jar/com/nimbusds/openid/connect/sdk/federation/entities/FederationMetadataType.java
* /opt/cola/permissions/1199843396_1630602842.72/oauth2-oidc-sdk-8-36-2-sources-
jar/com/nimbusds/openid/connect/sdk/federation/trust/constraints/EntityIDConstraint.java
* /opt/cola/permissions/1199843396_1630602842.72/oauth2-oidc-sdk-8-36-2-sources-
jar/com/nimbusds/openid/connect/sdk/federation/api/EntityListingRequest.java
* /opt/cola/permissions/1199843396_1630602842.72/oauth2-oidc-sdk-8-36-2-sources-
jar/com/nimbusds/openid/connect/sdk/federation/api/TrustNegotiationRequest.java
* /opt/cola/permissions/1199843396_1630602842.72/oauth2-oidc-sdk-8-36-2-sources-
jar/com/nimbusds/openid/connect/sdk/federation/policy/MetadataPolicyEntry.java
* /opt/cola/permissions/1199843396_1630602842.72/oauth2-oidc-sdk-8-36-2-sources-
jar/com/nimbusds/openid/connect/sdk/federation/policy/operations/DefaultPolicyOperationCombinationValidator.java
* /opt/cola/permissions/1199843396_1630602842.72/oauth2-oidc-sdk-8-36-2-sources-
jar/com/nimbusds/openid/connect/sdk/federation/trust/constraints/package-info.java
* /opt/cola/permissions/1199843396_1630602842.72/oauth2-oidc-sdk-8-36-2-sources-
jar/com/nimbusds/openid/connect/sdk/federation/policy/factories/DefaultRPMetadataPolicyFactory.java
* /opt/cola/permissions/1199843396_1630602842.72/oauth2-oidc-sdk-8-36-2-sources-
jar/com/nimbusds/openid/connect/sdk/federation/api/TrustNegotiationSuccessResponse.java
* /opt/cola/permissions/1199843396_1630602842.72/oauth2-oidc-sdk-8-36-2-sources-
jar/com/nimbusds/openid/connect/sdk/federation/api/EntityListingSuccessResponse.java
* /opt/cola/permissions/1199843396_1630602842.72/oauth2-oidc-sdk-8-36-2-sources-
jar/com/nimbusds/openid/connect/sdk/federation/trust/TrustChainRetriever.java
* /opt/cola/permissions/1199843396_1630602842.72/oauth2-oidc-sdk-8-36-2-sources-
jar/com/nimbusds/openid/connect/sdk/federation/policy/language/PolicyOperationApplication.java
* /opt/cola/permissions/1199843396_1630602842.72/oauth2-oidc-sdk-8-36-2-sources-
jar/com/nimbusds/openid/connect/sdk/federation/trust/constraints/SubtreeEntityIDConstraint.java
* /opt/cola/permissions/1199843396_1630602842.72/oauth2-oidc-sdk-8-36-2-sources-
jar/com/nimbusds/openid/connect/sdk/federation/trust/TrustChain.java
* /opt/cola/permissions/1199843396_1630602842.72/oauth2-oidc-sdk-8-36-2-sources-
jar/com/nimbusds/openid/connect/sdk/federation/policy/operations/package-info.java
* /opt/cola/permissions/1199843396_1630602842.72/oauth2-oidc-sdk-8-36-2-sources-
jar/com/nimbusds/openid/connect/sdk/rp/statement/InvalidSoftwareStatementException.java
* /opt/cola/permissions/1199843396_1630602842.72/oauth2-oidc-sdk-8-36-2-sources-
jar/com/nimbusds/openid/connect/sdk/claims/ClaimsSetRequest.java
* /opt/cola/permissions/1199843396_1630602842.72/oauth2-oidc-sdk-8-36-2-sources-
jar/com/nimbusds/openid/connect/sdk/federation/api/EntityListingResponse.java
Open Source Used In security_automation identity-guest-cs 529
No license file was found, but licenses were detected in source scan.

/*
 * oauth2-oidc-sdk
 *
 * Copyright 2012-2016, Connect2id Ltd and contributors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use
 * this file except in compliance with the License. You may obtain a copy of the
 * License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed
 * under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR
 * CONDITIONS OF ANY KIND, either express or implied. See the License for the
 * specific language governing permissions and limitations under the License.
 */

Found in path(s):
/*
 * oauth2-oidc-sdk
 *
 * Copyright 2012-2016, Connect2id Ltd and contributors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use
 * this file except in compliance with the License. You may obtain a copy of the
 * License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed
 * under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR
 * CONDITIONS OF ANY KIND, either express or implied. See the License for the
 * specific language governing permissions and limitations under the License.
 */
/**
 **
 * Creates a new authenticated pushed authorisation request for a
 * confidential client.
 *
 * @param uri          The URI of the token endpoint. May be
 *                     { @code null} if the {link #HTTPRequest}
 *                     method will not be used.
 * @param clientAuth   The client authentication. Must not be
 *                     { @code null}.
 * @param authzRequest The authorisation request. Must not be
 *                     { @code null}.
 *
 */

/**
 * oauth2-oidc-sdk
 *
 * Copyright 2012-2021, Connect2id Ltd and contributors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use
 * this file except in compliance with the License. You may obtain a copy of the
 * License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 */
* Unless required by applicable law or agreed to in writing, software distributed
* under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR
* CONDITIONS OF ANY KIND, either express or implied. See the License for the
* specific language governing permissions and limitations under the License.
*/
/**
* Authorisation code. Client authentication required only for
* confidential clients.
*/
/**
* Refresh token. Client authentication required only for confidential
* clients.
*/
/**
* Password. Client authentication required only for confidential
* clients.
*/

Found in path(s):
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8.36.2-sources-
jar/com/nimbusds/oauth2/sdk/GrantType.java

No license file was found, but licenses were detected in source scan.

/*
* oauth2-oidc-sdk
* *
* Copyright 2012-2016, Connect2id Ltd and contributors.
* *
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use
* this file except in compliance with the License. You may obtain a copy of the
* License at
* *
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed
* under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR
* CONDITIONS OF ANY KIND, either express or implied. See the License for the
* specific language governing permissions and limitations under the License.
*/
/**
* Token revocation request. Used to revoke an issued access or refresh token.
* *
* <p>Example token revocation request for a confidential client:
* *
* <pre>
* POST /revoke HTTP/1.1
* Host: server.example.com
Example token revocation request for a public client:

```
POST /revoke HTTP/1.1
Host: server.example.com
Content-Type: application/x-www-form-urlencoded

token=45ghiukldjahdnhzdauz&token_type_hint=refresh_token
```

Related specifications:

- OAuth 2.0 Token Revocation (RFC 7009), section 2.1.

---

Creates a new token revocation request for a confidential client.

```
/**
* Creates a new token revocation request for a confidential client.
*
* @param uri The URI of the token revocation endpoint. May be
*     {@code null} if the {@link #toHTTPRequest} method
*     will not be used.
* @param clientAuth The client authentication. Must not be
*     {@code null}.
* @param token The access or refresh token to revoke. Must not be
*     {@code null}.
*/
```

---

No license file was found, but licenses were detected in source scan.

```
* Copyright 2012-2021, Connect2id Ltd and contributors.
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use
* this file except in compliance with the License. You may obtain a copy of the
* License at
```
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed
* under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR
* CONDITIONS OF ANY KIND, either express or implied. See the License for the
* specific language governing permissions and limitations under the License.
*/

Found in path(s):
  */ /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/oauth2/sdk/ciba/CIBARequestAcknowledgement.java
  */ /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/oauth2/sdk/util/ListUtils.java
  */ /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/oauth2/sdk/ciba/CIBAErrorDelivery.java
  */ /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/oauth2/sdk/ciba/CIBASignedRequestClaimsSet.java
  */ /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/oauth2/sdk/ciba/CIBATokenDelivery.java
  */ /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/oauth2/sdk/ciba/CIBAPushCallback.java
  */ /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/oauth2/sdk/ciba/CIBAErro.java
  */ /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/oauth2/sdk/ciba/CIBAPingCallback.java

No license file was found, but licenses were detected in source scan.

/*
* oauth2-oidc-sdk
* 
* Copyright 2012-2016, Connect2id Ltd and contributors.
* 
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use
* this file except in compliance with the License. You may obtain a copy of the
* License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software distributed
* under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR
* CONDITIONS OF ANY KIND, either express or implied. See the License for the
* specific language governing permissions and limitations under the License.
*/
/**
* Client information. Encapsulates the registration and metadata details of
* an OAuth 2.0 client:
*
* <ul>
*   <li>The client identifier.
*   <li>The client metadata.
*   <li>The optional client secret for a confidential client.
*   <li>The optional registration URI and access token if dynamic client
    registration is permitted.
* </ul>

* Related specifications:

* <ul>
*   <li>OAuth 2.0 Dynamic Client Registration Protocol (RFC 7591), section
    3.2.1.
*   <li>OAuth 2.0 Dynamic Client Registration Management Protocol (RFC
    7592), section 3.
* </ul>

/**
// The client must by unambiguously public, else it is marked as confidential

Found in path(s):
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources-
  jar/com/nimbusds/oauth2/sdk/client/ClientInformation.java
No license file was found, but licenses were detected in source scan.

/*
* oauth2-oidc-sdk
*
* Copyright 2012-2016, Connect2id Ltd and contributors.
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use
* this file except in compliance with the License. You may obtain a copy of the
* License at
*
*    http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed
* under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR
* CONDITIONS OF ANY KIND, either express or implied. See the License for the
* specific language governing permissions and limitations under the License.
*/
/**
* Creates a new request object POST request where the parameters are
* submitted as plain JSON object, and the client authenticates by
* means of mutual TLS. TLS also ensures the integrity and
* confidentiality of the request parameters. This method is not
* standard.
*@
* @param uri The URI of the request object endpoint. May
be [code null] if the
{@link #toHTTPRequest} method will not be
used.
@param tlsClientAuth The mutual TLS client authentication. Must
not be [code null].
@param requestJSONObject The request parameters as plain JSON
object. Must not be [code null].
*/

Found in path(s):
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources-
jar/com/nimbusds/oauth2/sdk/RequestObjectPOSTRequest.java
No license file was found, but licenses were detected in source scan.

/*
* oauth2-oidc-sdk
*
* Copyright 2012-2016, Connect2id Ltd and contributors.
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use
* this file except in compliance with the License. You may obtain a copy of the
* License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed
* under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR
* CONDITIONS OF ANY KIND, either express or implied. See the License for the
* specific language governing permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources-
jar/com/nimbusds/oauth2/sdk/auth/ClientSecretJWT.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources-
jar/com/nimbusds/oauth2/sdk/jose/package-info.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources-
jar/com/nimbusds/oauth2/sdk/TokenIntrospectionSuccessResponse.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources-
jar/com/nimbusds/openid/connect/sdk/rp/OIDCClientUpdateRequest.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources-
jar/com/nimbusds/openid/connect/sdk/http/ServletUtils.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources-
jar/com/nimbusds/openid/connect/sdk/token/BearerAccessToken.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources-
jar/com/nimbusds/openid/connect/sdk/op/AuthenticationRequestDetector.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources-
jar/com/nimbusds/openid/connect/sdk/federation/policy/language/package-info.java
Open Source Used In security_automation identity-guest-cs  539

* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/oauth2/sdk/ciba/BackChannelTokenDeliveryMode.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/openid/connect/sdk/federation/policy/package-info.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/openid/connect/sdk/OIDCResponseTypeValue.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/openid/connect/sdk/federation/policy/operations/DefaultOperation.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/openid/connect/sdk/ProtectedResourceRequest.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/openid/connect/sdk/claims/ClaimsTransport.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/openid/connect/sdk/assertions/package-info.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/openid/connect/sdk/AMR.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/openid/connect/sdk/Address.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/openid/connect/sdk/AuthenticationResponseParser.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/openid/connect/sdk/util/MultivaluedMapUtils.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/openid/connect/sdk/RequestObjectPOSTResponse.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/openid/connect/sdk/ParseException.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/openid/connect/sdk/ClientAuthenticationMethod.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources-jar/com/nimbusds/oauth2/sdk/token/RefreshToken.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources-jar/com/nimbusds/oauth2/sdk/token/AccessToken.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources-jar/com/nimbusds/openid/connect/sdk/op/AuthenticationRequestResolver.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources-jar/com/nimbusds/oauth2/sdk/RefreshTokenGrant.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources-jar/com/nimbusds/openid/connect/sdk/LogoutRequest.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources-jar/com/nimbusds/oauth2/sdk/assertions/jwt/package-info.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources-jar/com/nimbusds/oauth2/sdk/auth/verifier/ClientCredentialsSelector.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources-jar/com/nimbusds/oauth2/sdk/client/package-info.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources-jar/com/nimbusds/oauth2/sdk/util/URLUtils.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources-jar/com/nimbusds/oauth2/sdk/util/JSONUtils.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources-jar/com/nimbusds/oauth2/sdk/util/URIUtils.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources-jar/com/nimbusds/oauth2/sdk/http/HTTPRequest.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources-jar/com/nimbusds/oauth2/sdk/PushedAuthorizationResponse.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources-jar/com/nimbusds/openid/connect/sdk/federation/entities/FederationEntityMetadata.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources-jar/com/nimbusds/oauth2/sdk/ResourceOwnerPasswordCredentialsGrant.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources-jar/com/nimbusds/openid/connect/sdk/token/OIDCTokens.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources-jar/com/nimbusds/oauth2/sdk/SAML2BearerGrant.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources-jar/com/nimbusds/openid/connect/sdk/UserInfoResponse.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources-jar/com/nimbusds/oauth2/sdk/client/RegistrationError.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources-jar/com/nimbusds/oauth2/sdk/auth/package-info.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources-jar/com/nimbusds/openid/connect/sdk/federation/entities/EntityStatementClaimsSet.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources-jar/com/nimbusds/openid/connect/sdk/OIDCResponseTypeValidator.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources-jar/com/nimbusds/oauth2/sdk/token/AccessTokenType.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources-jar/com/nimbusds/openid/connect/sdk/federation/policy/operations/ValueOperation.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources-jar/com/nimbusds/oauth2/sdk/Response.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources-jar/com/nimbusds/openid/connect/sdk/Prompt.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources-jar/com/nimbusds/openid/connect/sdk/claims/HashClaim.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources-jar/com/nimbusds/openid/connect/sdk/federation/trust/marks/TrustMarkClaimsSet.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources-jar/com/nimbusds/openid/connect/sdk/assurance/IdentityVerification.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources-jar/com/nimbusds/oauth2/sdk/id/SoftwareID.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources-jar/com/nimbusds/oauth2/sdk/ErrorObject.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources-jar/com/nimbusds/oauth2/sdk/device/DeviceAuthorizationRequest.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/oauth2/sdk/Scope.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/oauth2/sdk/SuccessResponse.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/openid/connect/sdk/rp/OIDCClientMetadata.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/openid/connect/sdk/fe/federation/policy/operations/EssentialOperation.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/openid/connect/sdk/OIDCTokenResponse.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/oauth2/sdk/auth/PlainClientSecret.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/oauth2/sdk/auth/verifier/Context.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/oauth2/sdk/http/HTTPResponse.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/oauth2/sdk/id/State.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/oauth2/sdk/util/tls/package-info.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/openid/connect/sdk/federation/policy/operations/AddOperation.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/openid/connect/sdk/assertions/saml2/SAML2AssertionDetailsVerifier.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/openid/connect/sdk/assurance/VerificationProcess.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/openid/connect/sdk/claims/ExternalClaims.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/openid/connect/sdk/assurance/evidences/UtilityBillEvidence.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/openid/connect/sdk/claims/CommonClaimsSet.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/oauth2/sdk/token/BearerTokenError.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/oauth2/sdk/ciba/CIBAResponse.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/oauth2/sdk/jarm/JARMUtils.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/oauth2/sdk/AuthorizationResponse.java

Open Source Used In security_automation identity-guest-cs 544
* /opt/cola/permits/1199843396_1630602842.72/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/oauth2/sdk/assertions/saml2/BadSAML2AssertionException.java
* /opt/cola/permits/1199843396_1630602842.72/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/openid/connect/sdk/id/SectorID.java
* /opt/cola/permits/1199843396_1630602842.72/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/oauth2/sdk/AssertionGrant.java
* /opt/cola/permits/1199843396_1630602842.72/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/openid/connect/sdk/id/AssertionGrant.java
* /opt/cola/permits/1199843396_1630602842.72/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/openid/connect/sdk/id/package-info.java
* /opt/cola/permits/1199843396_1630602842.72/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/openid/connect/sdk/federation/package-info.java
* /opt/cola/permits/1199843396_1630602842.72/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/openid/connect/sdk/federation/policy/language/PolicyViolationException.java
* /opt/cola/permits/1199843396_1630602842.72/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/openid/connect/sdk/validators/AccessTokenValidator.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/openid/connect/sdk/Display.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/oauth2/sdk/token/Token.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/oauth2/sdk/auth/ClientSecretPost.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/oauth2/sdk/jarm/JARMClaimsVerifier.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/oauth2/sdk/pkce/package-info.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/oauth2/sdk/util/MapUtils.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/oauth2/sdk/util/JWTClaimsSetUtils.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/oauth2/sdk/AuthorizationCode.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/oauth2/sdk/PushedAuthorizationSuccessResponse.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/oauth2/sdk/id/SoftwareVersion.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/openid/connect/sdk/rp/OIDCClientRegistrationResponseParser.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/openid/connect/sdk/federation/policy/language/OperationName.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/oauth2/sdk/id/Actor.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/oauth2/sdk/token/package-info.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/oauth2/sdk/device/DeviceAuthorizationErrorResponse.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/oauth2/sdk/device/DeviceAuthorizationResponse.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/oauth2/sdk/TokenIntrospectionResponse.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/oauth2/sdk/auth/ClientAuthentication.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/oauth2/sdk/id/Audience.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/openid/connect/sdk/federation/policy/operations/OneOfOperation.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/oauth2/sdk/auth/ClientSecretBasic.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources.jar/com/nimbusds/oauth2/sdk/as/AuthorizationServerMetadata.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources-jar/com/nimbusds/oauth2/sdk/client/ClientRegistrationRequest.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources-jar/com/nimbusds/oauth2/sdk/as/AuthorizationServerConfigurationRequest.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources-jar/com/nimbusds/oauth2/sdk/auth/verifier/JWTAuthenticationClaimsSetVerifier.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources-jar/com/nimbusds/oauth2/sdk/openid/connect/sdk/op/ACRRequest.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources-jar/com/nimbusds/oauth2/sdk/token/Tokens.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources-jar/com/nimbusds/oauth2/sdk/id/AuthorizedParty.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources-jar/com/nimbusds/openid/connect/sdk/package-info.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources-jar/com/nimbusds/oauth2/sdk/auth/PrivateKeyJWT.java
* /opt/cola/permits/1199843396_1630602842.72/0/oauth2-oidc-sdk-8-36-2-sources-jar/com/nimbusds/openid/connect/sdk/federation/policy/language/NumberConfiguration.java

1.100 okio 1.17.2

1.101 json-schema-validator 1.0.57

1.101.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2020 Network New Technologies Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
~ distributed under the License is distributed on an "AS IS" BASIS,
~ WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
~ See the License for the specific language governing permissions and
~ limitations under the License.

Found in path(s):
No license file was found, but licenses were detected in source scan.

/*
 * Licensed to the Apache Software Foundation (ASF) under one or more
 * contributor license agreements. See the NOTICE file distributed with
 * this work for additional information regarding copyright ownership.
 * The ASF licenses this file to You under the Apache License, Version 2.0
 * (the "License"); you may not use this file except in compliance with
 * the License. You may obtain a copy of the License at
 *
 *      http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1201879935_1631304744.93/0/json-schema-validator-1-0-57-sources-jar/com/networknt/schema/format/RegexValidator.java
* /opt/cola/permits/1201879935_1631304744.93/0/json-schema-validator-1-0-57-sources-jar/com/networknt/schema/format/InetAddressValidator.java
* /opt/cola/permits/1201879935_1631304744.93/0/json-schema-validator-1-0-57-sources-jar/com/networknt/schema/format/EmailValidator.java
* /opt/cola/permits/1201879935_1631304744.93/0/json-schema-validator-1-0-57-sources-jar/com/networknt/schema/format/DomainValidator.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2016 Network New Technologies Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *      http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
  * /opt/cola/permits/1201879935_1631304744.93/0/json-schema-validator-1-0-57-sources-
    jar/com/networknt/schema/JsonType.java
  * /opt/cola/permits/1201879935_1631304744.93/0/json-schema-validator-1-0-57-sources-
    jar/com/networknt/schema/uri/URIFactory.java
  * /opt/cola/permits/1201879935_1631304744.93/0/json-schema-validator-1-0-57-sources-
    jar/com/networknt/schema/ValidatorTypeCode.java
  * /opt/cola/permits/1201879935_1631304744.93/0/json-schema-validator-1-0-57-sources-
    jar/com/networknt/schema/NonValidationKeyword.java
  * /opt/cola/permits/1201879935_1631304744.93/0/json-schema-validator-1-0-57-sources-
    jar/com/networknt/schema/MinLengthValidator.java
  * /opt/cola/permits/1201879935_1631304744.93/0/json-schema-validator-1-0-57-sources-
    jar/com/networknt/schema/Format.java
  * /opt/cola/permits/1201879935_1631304744.93/0/json-schema-validator-1-0-57-sources-
    jar/com/networknt/schema/AbstractFormat.java
  * /opt/cola/permits/1201879935_1631304744.93/0/json-schema-validator-1-0-57-sources-
    jar/com/networknt/schema/PropertiesValidator.java
  * /opt/cola/permits/1201879935_1631304744.93/0/json-schema-validator-1-0-57-sources-
    jar/com/networknt/schema/MinItemsValidator.java
  * /opt/cola/permits/1201879935_1631304744.93/0/json-schema-validator-1-0-57-sources-
    jar/com/networknt/schema/MaximumValidator.java
  * /opt/cola/permits/1201879935_1631304744.93/0/json-schema-validator-1-0-57-sources-
    jar/com/networknt/schema/JsonValidator.java
  * /opt/cola/permits/1201879935_1631304744.93/0/json-schema-validator-1-0-57-sources-
    jar/com/networknt/schema/JsonSchemaFactory.java
  * /opt/cola/permits/1201879935_1631304744.93/0/json-schema-validator-1-0-57-sources-
    jar/com/networknt/schema/AnyOfValidator.java
  * /opt/cola/permits/1201879935_1631304744.93/0/json-schema-validator-1-0-57-sources-
    jar/com/networknt/schema/ExclusiveMaximumValidator.java
  * /opt/cola/permits/1201879935_1631304744.93/0/json-schema-validator-1-0-57-sources-
    jar/com/networknt/schema/ContainsValidator.java
  * /opt/cola/permits/1201879935_1631304744.93/0/json-schema-validator-1-0-57-sources-
    jar/com/networknt/schema/UUIDValidator.java
  * /opt/cola/permits/1201879935_1631304744.93/0/json-schema-validator-1-0-57-sources-
    jar/com/networknt/schema/MaxPropertiesValidator.java
  * /opt/cola/permits/1201879935_1631304744.93/0/json-schema-validator-1-0-57-sources-
    jar/com/networknt/schema/AbstractKeyword.java
  * /opt/cola/permits/1201879935_1631304744.93/0/json-schema-validator-1-0-57-sources-
    jar/com/networknt/schema/uri/URLFactory.java
  * /opt/cola/permits/1201879935_1631304744.93/0/json-schema-validator-1-0-57-sources-
    jar/com/networknt/schema/MinPropertiesValidator.java
  * /opt/cola/permits/1201879935_1631304744.93/0/json-schema-validator-1-0-57-sources-
1.102 jackson-datatype-jdk8 2.13.0
1.102.1 Available under license:
    Apache-2.0

1.103 objenesis 3.1
1.103.1 Available under license:
    // ------------------------------------------------------------------
    // NOTICE file corresponding to the section 4d of The Apache License,
    // Version 2.0, in this case for Objenesis
    // ------------------------------------------------------------------
    Objenesis
    Copyright 2006-2019 Joe Walnes, Henri Tremblay, Leonardo Mesquita

    Apache License
    Version 2.0, January 2004
    http://www.apache.org/licenses/

    TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

    "License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

    "Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

    "Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

    "You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

    "Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

    "Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
    Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
    stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
    that You distribute, all copyright, patent, trademark, and
    attribution notices from the Source form of the Work,
    excluding those notices that do not pertain to any part of
    the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.
END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.104 activation 1.1

1.104.1 Available under license:

Found license 'Common Development and Distribution License (CDDL) 1.1' in "The contents of this file are subject to the terms of the Common Development and Distribution License (the "License"). You may not use this file except Copyright 2001-2005 Sun Microsystems, Inc. All Rights Reserved.'

Found license 'Common Development and Distribution License (CDDL) 1.1' in "* The contents of this file are subject to the terms * of the Common Development and Distribution License * (the "License"). You may not use this file except * Copyright 2002-2005 Sun Microsystems, Inc. All Rights Reserved.'

Found license 'Common Development and Distribution License (CDDL) 1.1' in "* The contents of this file are subject to the terms * of the Common Development and Distribution License * (the "License"). You may not use this file except * Copyright 1997-2005 Sun Microsystems, Inc. All Rights Reserved.'

1.105 spring-cloud-commons 3.0.4
1.105.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2013-2020 the original author or authors.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 *   https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Find in path(s):
* /opt/cola/permits/1222421618_1635579121.68/0/spring-cloud-commons-3-0-4-sources-jar/org/springframework/cloud/client/circuitbreaker/CircuitBreaker.java
* /opt/cola/permits/1222421618_1635579121.68/0/spring-cloud-commons-3-0-4-sources-jar/org/springframework/cloud/client/circuitbreaker/ReactiveCircuitBreaker.java
* /opt/cola/permits/1222421618_1635579121.68/0/spring-cloud-commons-3-0-4-sources-jar/org/springframework/cloud/client/circuitbreaker/NoFallbackAvailableException.java
* /opt/cola/permits/1222421618_1635579121.68/0/spring-cloud-commons-3-0-4-sources-jar/org/springframework/cloud/client/circuitbreaker/AbstractCircuitBreakerFactory.java
* /opt/cola/permits/1222421618_1635579121.68/0/spring-cloud-commons-3-0-4-sources-jar/org/springframework/cloud/client/circuitbreaker/ConfigBuilder.java
* /opt/cola/permits/1222421618_1635579121.68/0/spring-cloud-commons-3-0-4-sources-jar/org/springframework/cloud/client/circuitbreaker/Customizer.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015-2021 the original author or authors.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 *   https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1222421618_1635579121.68/0/spring-cloud-commons-3-0-4-sources-jar/org/springframework/cloud/commons/ConfigDataMissingEnvironmentPostProcessor.java
No license file was found, but licenses were detected in source scan.

<name>Apache License, Version 2.0</name>
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at
distributed under the License is distributed on an "AS IS" BASIS,

Found in path(s):
* /opt/cola/permits/1222421618_1635579121.68/0/spring-cloud-commons-3-0-4-sources-jar/META-INF/maven/org.springframework.cloud/spring-cloud-commons/pom.xml
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2019-2020 the original author or authors.
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1222421618_1635579121.68/0/spring-cloud-commons-3-0-4-sources-jar/org/springframework/cloud/client/ConditionalOnBlockingDiscoveryEnabled.java
* /opt/cola/permits/1222421618_1635579121.68/0/spring-cloud-commons-3-0-4-sources-jar/org/springframework/cloud/client/ConditionalOnDiscoveryEnabled.java
* /opt/cola/permits/1222421618_1635579121.68/0/spring-cloud-commons-3-0-4-sources-jar/org/springframework/cloud/client/ConditionalOnReactiveDiscoveryEnabled.java
* /opt/cola/permits/1222421618_1635579121.68/0/spring-cloud-commons-3-0-4-sources-jar/org/springframework/cloud/client/ConditionalOnDiscoveryHealthIndicatorEnabled.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2012-2020 the original author or authors.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * https://www.apache.org/licenses/LICENSE-2.0
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1222421618_1635579121.68/0/spring-cloud-commons-3-0-4-sources-jar/org/springframework/cloud/client/loadbalancer/LoadBalancerProperties.java
* /opt/cola/permits/1222421618_1635579121.68/0/spring-cloud-commons-3-0-4-sources-jar/org/springframework/cloud/client/discovery/composite/reactive/ReactiveCompositeDiscoveryClient.java
* /opt/cola/permits/1222421618_1635579121.68/0/spring-cloud-commons-3-0-4-sources-jar/org/springframework/cloud/client/loadbalancer/LoadBalancedRecoveryCallback.java
* /opt/cola/permits/1222421618_1635579121.68/0/spring-cloud-commons-3-0-4-sources-jar/org/springframework/cloud/commons/util/InetUtils.java
* /opt/cola/permits/1222421618_1635579121.68/0/spring-cloud-commons-3-0-4-sources-jar/org/springframework/cloud/client/loadbalancer/LoadBalancerRequest.java
* /opt/cola/permits/1222421618_1635579121.68/0/spring-cloud-commons-3-0-4-sources-jar/org/springframework/cloud/commons/httpclient/DefaultApacheHttpClientConnectionManagerFactory.java
* /opt/cola/permits/1222421618_1635579121.68/0/spring-cloud-commons-3-0-4-sources-jar/org/springframework/cloud/client/discovery/composite/reactive/DeferringLoadBalancerExchangeFilterFunction.java
* /opt/cola/permits/1222421618_1635579121.68/0/spring-cloud-commons-3-0-4-sources-jar/org/springframework/cloud/client/serviceregistry/ServiceRegistryAutoConfiguration.java
jar/org/springframework/cloud/client/DefaultServiceInstance.java
* /opt/cola/permits/1222421618_1635579121.68/0/spring-cloud-commons-3-0-4-sources-
jar/org/springframework/cloud/client/loadbalancer/Request.java
* /opt/cola/permits/1222421618_1635579121.68/0/spring-cloud-commons-3-0-4-sources-
jar/org/springframework/cloud/client/loadbalancer/CompatibilityNotMetException.java
* /opt/cola/permits/1222421618_1635579121.68/0/spring-cloud-commons-3-0-4-sources-
jar/org/springframework/cloud/client/discovery/health/DiscoveryClientHealthIndicator.java
* /opt/cola/permits/1222421618_1635579121.68/0/spring-cloud-commons-3-0-4-sources-
jar/org/springframework/cloud/client/loadbalancer/AsyncLoadBalancerInterceptor.java
* /opt/cola/permits/1222421618_1635579121.68/0/spring-cloud-commons-3-0-4-sources-
jar/org/springframework/cloud/client/loadbalancer/reactive/LoadBalancerWebClientBuilderBeanPostProcessor.java
* /opt/cola/permits/1222421618_1635579121.68/0/spring-cloud-commons-3-0-4-sources-
jar/org/springframework/cloud/commons/httpclient/OkHttpClientFactory.java
* /opt/cola/permits/1222421618_1635579121.68/0/spring-cloud-commons-3-0-4-sources-
jar/org/springframework/cloud/client/discovery/health/reactive/ReactiveDiscoveryCompositeHealthContributor.java
* /opt/cola/permits/1222421618_1635579121.68/0/spring-cloud-commons-3-0-4-sources-
jar/org/springframework/cloud/client/loadbalancer/reactive/LoadBalancerRetryContext.java
* /opt/cola/permits/1222421618_1635579121.68/0/spring-cloud-commons-3-0-4-sources-
jar/org/springframework/cloud/client/loadbalancer/DefaultRequestContext.java
* /opt/cola/permits/1222421618_1635579121.68/0/spring-cloud-commons-3-0-4-sources-
jar/org/springframework/cloud/client/discovery/health/reactive/ReactiveDiscoveryClientHealthIndicator.java
* /opt/cola/permits/1222421618_1635579121.68/0/spring-cloud-commons-3-0-4-sources-
jar/org/springframework/cloud/client/loadbalancer/EmptyResponse.java
* /opt/cola/permits/1222421618_1635579121.68/0/spring-cloud-commons-3-0-4-sources-
jar/org/springframework/cloud/client/common/ServiceInstanceChooser.java
* /opt/cola/permits/1222421618_1635579121.68/0/spring-cloud-commons-3-0-4-sources-
jar/org/springframework/cloud/client/configuration/CompatibilityVerifier.java
* /opt/cola/permits/1222421618_1635579121.68/0/spring-cloud-commons-3-0-4-sources-
jar/org/springframework/cloud/client/discovery/health/reactive/ReactiveDiscoveryClientHealthIndicator.java
* /opt/cola/permits/1222421618_1635579121.68/0/spring-cloud-commons-3-0-4-sources-
jar/org/springframework/cloud/client/loadbalancer/ServiceInstanceChooser.java
* /opt/cola/permits/1222421618_1635579121.68/0/spring-cloud-commons-3-0-4-sources-
Open Source Used In security_automation identity-guest-cs 566

* /opt/cola/permits/1222421618_1635579121.68/0/spring-cloud-commons-3-0-4-sources-jar/org/springframework/cloud/client/discovery/simple/reactive/SimpleReactiveDiscoveryProperties.java
* /opt/cola/permits/1222421618_1635579121.68/0/spring-cloud-commons-3-0-4-sources-jar/org/springframework/cloud/client/SpringCloudApplication.java
* /opt/cola/permits/1222421618_1635579121.68/0/spring-cloud-commons-3-0-4-sources-jar/org/springframework/cloud/client/loadbalancer/RestTemplateCustomizer.java
* /opt/cola/permits/1222421618_1635579121.68/0/spring-cloud-commons-3-0-4-sources-jar/org/springframework/cloud/client/loadbalancer/reactive/LoadBalancerClientRequestTransformer.java
* /opt/cola/permits/1222421618_1635579121.68/0/spring-cloud-commons-3-0-4-sources-jar/org/springframework/cloud/commons/httpclient/DefaultOkHttpClientFactory.java
* /opt/cola/permits/1222421618_1635579121.68/0/spring-cloud-commons-3-0-4-sources-jar/org/springframework/cloud/client/loadbalancer/LoadBalancerRequestTransformer.java
* /opt/cola/permits/1222421618_1635579121.68/0/spring-cloud-commons-3-0-4-sources-jar/org/springframework/cloud/client/loadbalancer/DefaultResponse.java
* /opt/cola/permits/1222421618_1635579121.68/0/spring-cloud-commons-3-0-4-sources-jar/org/springframework/cloud/client/loadbalancer/reactive/LoadBalancedExchangeFilterFunction.java
* /opt/cola/permits/1222421618_1635579121.68/0/spring-cloud-commons-3-0-4-sources-jar/org/springframework/cloud/client/discovery/health/reactive/ReactiveDiscoveryHealthIndicator.java
* /opt/cola/permits/1222421618_1635579121.68/0/spring-cloud-commons-3-0-4-sources-jar/org/springframework/cloud/client/servicecatalog/AbstractAutoServiceRegistration.java
* /opt/cola/permits/1222421618_1635579121.68/0/spring-cloud-commons-3-0-4-sources-jar/org/springframework/cloud/client/hypermedia/RemoteResource.java
* /opt/cola/permits/1222421618_1635579121.68/0/spring-cloud-commons-3-0-4-sources-jar/org/springframework/cloud/client/discovery/event/ParentHeartbeatEvent.java
* /opt/cola/permits/1222421618_1635579121.68/0/spring-cloud-commons-3-0-4-sources-jar/org/springframework/cloud/client/loadbalancer/ServiceRequestWrapper.java
* /opt/cola/permits/1222421618_1635579121.68/0/spring-cloud-commons-3-0-4-sources-jar/org/springframework/cloud/client/discovery/EnableDiscoveryClientImportSelector.java
* /opt/cola/permits/1222421618_1635579121.68/0/spring-cloud-commons-3-0-4-sources-jar/org/springframework/cloud/client/circuitbreaker/EnableCircuitBreakerImportSelector.java
* /opt/cola/permits/1222421618_1635579121.68/0/spring-cloud-commons-3-0-4-sources-jar/org/springframework/cloud/client/discovery/composite/CompositeDiscoveryClientAutoConfiguration.java
* /opt/cola/permits/1222421618_1635579121.68/0/spring-cloud-commons-3-0-4-sources-jar/org/springframework/cloud/client/actuator/HasFeatures.java
* /opt/cola/permits/1222421618_1635579121.68/0/spring-cloud-commons-3-0-4-sources-jar/org/springframework/cloud/client/loadbalancer/CompletionContext.java
* /opt/cola/permits/1222421618_1635579121.68/0/spring-cloud-commons-3-0-4-sources-jar/org/springframework/cloud/client/hypermedia/TraversalDefinition.java
* /opt/cola/permits/1222421618_1635579121.68/0/spring-cloud-commons-3-0-4-sources-jar/org/springframework/cloud/client/discovery/health/DiscoveryCompositeHealthContributor.java
* /opt/cola/permits/1222421618_1635579121.68/0/spring-cloud-commons-3-0-4-sources-jar/org/springframework/cloud/client/discovery/event/InstanceRegisteredEvent.java
* /opt/cola/permits/1222421618_1635579121.68/0/spring-cloud-commons-3-0-4-sources-jar/org/springframework/cloud/client/loadbalancer/LoadBalancerAutoConfiguration.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2012-2015 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1222421618_1635579121.68/0/spring-cloud-commons-3-0-4-sources-
  jar/org/springframework/cloud/commons/security/ResourceServerTokenRelayAutoConfiguration.java
* /opt/cola/permits/1222421618_1635579121.68/0/spring-cloud-commons-3-0-4-sources-
  jar/org/springframework/cloud/commons/security/AccessTokenContextRelay.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2013-2021 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1222421618_1635579121.68/0/spring-cloud-commons-3-0-4-sources-
  jar/org/springframework/cloud/commons/publisher/FluxFirstNonEmptyEmitting.java
* /opt/cola/permits/1222421618_1635579121.68/0/spring-cloud-commons-3-0-4-sources-
  jar/org/springframework/cloud/commons/publisher/CloudFlux.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2017-2020 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* Unlicensed required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2012-2021 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*      https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

No license file was found, but licenses were detected in source scan.

1.106 msal4j 1.2.0
1.106.1 Available under license:
No license file was found, but licenses were detected in source scan.

// Copyright (c) Microsoft Corporation. All rights reserved.
// Licensed under the MIT License.
/**
 * Factory for creating client credentials used in confidential client flows
 */

Found in path(s):
* /opt/cola/permits/1256060467_1642635517.64/0/msal4j-1-2-0-sources-
  jar/com/microsoft/aad/msal4j/ClientCredentialFactory.java
No license file was found, but licenses were detected in source scan.

// Copyright (c) Microsoft Corporation. All rights reserved.
// Licensed under the MIT License.
/**
 * Object containing parameters for authorization code flow. Can be used as parameter to
 * [PublicClientApplication#acquireToken(AuthorizationCodeParameters)] or to
 * [ConfidentialClientApplication#acquireToken(AuthorizationCodeParameters)]
 */

Found in path(s):
* /opt/cola/permits/1256060467_1642635517.64/0/msal4j-1-2-0-sources-
  jar/com/microsoft/aad/msal4j/AuthorizationCodeParameters.java
No license file was found, but licenses were detected in source scan.

// Copyright (c) Microsoft Corporation. All rights reserved.
// Licensed under the MIT License.
/**
 * Object containing parameters for On-Behalf-Of flow. Can be used as parameter to
 * [ConfidentialClientApplication#acquireToken(OnBehalfOfParameters)]
 */

Found in path(s):
* /opt/cola/permits/1256060467_1642635517.64/0/msal4j-1-2-0-sources-
  jar/com/microsoft/aad/msal4j/OnBehalfOfParameters.java
No license file was found, but licenses were detected in source scan.

// Copyright (c) Microsoft Corporation. All rights reserved.
// Licensed under the MIT License.
/**
 * Object containing parameters for client credential flow. Can be used as parameter to
 * [ConfidentialClientApplication#acquireToken(ClientCredentialParameters)]
 */

Found in path(s):
* /opt/cola/permits/1256060467_1642635517.64/0/msal4j-1-2-0-sources-

No license file was found, but licenses were detected in source scan.

// Copyright (c) Microsoft Corporation. All rights reserved.
// Licensed under the MIT License.

Found in path(s):
* /opt/cola/permits/1256060467_1642635517.64/0/msal4j-1-2-0-sources-jar/com/microsoft/aad/msal4j/TokenCache.java
* /opt/cola/permits/1256060467_1642635517.64/0/msal4j-1-2-0-sources-jar/com/microsoft/aad/msal4j/MexParser.java
* /opt/cola/permits/1256060467_1642635517.64/0/msal4j-1-2-0-sources-jar/com/microsoft/aad/msal4j/OAuthAuthorizationGrant.java
* /opt/cola/permits/1256060467_1642635517.64/0/msal4j-1-2-0-sources-jar/com/microsoft/aad/msal4j/HttpUtils.java
* /opt/cola/permits/1256060467_1642635517.64/0/msal4j-1-2-0-sources-jar/com/microsoft/aad/msal4j/IdTokenCacheEntity.java
* /opt/cola/permits/1256060467_1642635517.64/0/msal4j-1-2-0-sources-jar/com/microsoft/aad/msal4j/JsonHelper.java
* /opt/cola/permits/1256060467_1642635517.64/0/msal4j-1-2-0-sources-jar/com/microsoft/aad/msal4j/Account.java
* /opt/cola/permits/1256060467_1642635517.64/0/msal4j-1-2-0-sources-jar/com/microsoft/aad/msal4j/ApiEvent.java
* /opt/cola/permits/1256060467_1642635517.64/0/msal4j-1-2-0-sources-jar/com/microsoft/aad/msal4j/InstanceDiscoveryResponse.java
* /opt/cola/permits/1256060467_1642635517.64/0/msal4j-1-2-0-sources-jar/com/microsoft/aad/msal4j/AbstractMsalAuthorizationGrant.java
* /opt/cola/permits/1256060467_1642635517.64/0/msal4j-1-2-0-sources-jar/com/microsoft/aad/msal4j/ITokenCacheAccessAspect.java
* /opt/cola/permits/1256060467_1642635517.64/0/msal4j-1-2-0-sources-jar/com/microsoft/aad/msal4j/TokenRequestExecutor.java
* /opt/cola/permits/1256060467_1642635517.64/0/msal4j-1-2-0-sources-jar/com/microsoft/aad/msal4j/TelemetryConstants.java
* /opt/cola/permits/1256060467_1642635517.64/0/msal4j-1-2-0-sources-jar/com/microsoft/aad/msal4j/AuthenticationResultSupplier.java
* /opt/cola/permits/1256060467_1642635517.64/0/msal4j-1-2-0-sources-jar/com/microsoft/aad/msal4j/ClientCredentialRequest.java
* /opt/cola/permits/1256060467_1642635517.64/0/msal4j-1-2-0-sources-jar/com/microsoft/aad/msal4j/UserDiscoveryRequest.java
* /opt/cola/permits/1256060467_1642635517.64/0/msal4j-1-2-0-sources-jar/com/microsoft/aad/msal4j/BindingPolicy.java
* /opt/cola/permits/1256060467_1642635517.64/0/msal4j-1-2-0-sources-jar/com/microsoft/aad/msal4j/WSTrustResponse.java
* /opt/cola/permits/1256060467_1642635517.64/0/msal4j-1-2-0-sources-jar/com/microsoft/aad/msal4j/LogHelper.java
* /opt/cola/permits/1256060467_1642635517.64/0/msal4j-1-2-0-sources-jar/com/microsoft/aad/msal4j/ITokenCacheAccessContext.java
* /opt/cola/permits/1256060467_1642635517.64/0/msal4j-1-2-0-sources-
jar/com/microsoft/aad/msal4j/IHttpResponse.java
* /opt/cola/permits/1256060467_1642635517.64/0/msal4j-1-2-0-sources-jar/com/microsoft/aad/msal4j/HttpResponse.java
* /opt/cola/permits/1256060467_1642635517.64/0/msal4j-1-2-0-sources-jar/com/microsoft/aad/msal4j/HttpRequest.java
* /opt/cola/permits/1256060467_1642635517.64/0/msal4j-1-2-0-sources-jar/com/microsoft/aad/msal4j/AadInstanceDiscovery.java
* /opt/cola/permits/1256060467_1642635517.64/0/msal4j-1-2-0-sources-jar/com/microsoft/aad/msal4j/ITelemetry.java
* /opt/cola/permits/1256060467_1642635517.64/0/msal4j-1-2-0-sources-jar/com/microsoft/aad/msal4j/ITelemetryManager.java
* /opt/cola/permits/1256060467_1642635517.64/0/msal4j-1-2-0-sources-jar/com/microsoft/aad/msal4j/IClientAssertion.java
* /opt/cola/permits/1256060467_1642635517.64/0/msal4j-1-2-0-sources-jar/com/microsoft/aad/msal4j/IClientCertificate.java
* /opt/cola/permits/1256060467_1642635517.64/0/msal4j-1-2-0-sources-jar/com/microsoft/aad/msal4j/AuthorizationCodeRequest.java
* /opt/cola/permits/1256060467_1642635517.64/0/msal4j-1-2-0-sources-jar/com/microsoft/aad/msal4j/RefreshTokenCacheEntity.java
* /opt/cola/permits/1256060467_1642635517.64/0/msal4j-1-2-0-sources-jar/com/microsoft/aad/msal4j/AuthenticationResult.java
* /opt/cola/permits/1256060467_1642635517.64/0/msal4j-1-2-0-sources-jar/com/microsoft/aad/msal4j/SAML11BearerGrant.java
* /opt/cola/permits/1256060467_1642635517.64/0/msal4j-1-2-0-sources-jar/com/microsoft/aad/msal4j/IntegratedWindowsAuthorizationGrant.java
* /opt/cola/permits/1256060467_1642635517.64/0/msal4j-1-2-0-sources-jar/com/microsoft/aad/msal4j/Constants.java
* /opt/cola/permits/1256060467_1642635517.64/0/msal4j-1-2-0-sources-jar/com/microsoft/aad/msal4j/MsalClientException.java
* /opt/cola/permits/1256060467_1642635517.64/0/msal4j-1-2-0-sources-jar/com/microsoft/aad/msal4j/B2CAuthority.java
* /opt/cola/permits/1256060467_1642635517.64/0/msal4j-1-2-0-sources-jar/com/microsoft/aad/msal4j/AuthenticationErrorMessage.java
* /opt/cola/permits/1256060467_1642635517.64/0/msal4j-1-2-0-sources-jar/com/microsoft/aad/msal4j/AcquireTokenSilentSupplier.java
* /opt/cola/permits/1256060467_1642635517.64/0/msal4j-1-2-0-sources-jar/com/microsoft/aad/msal4j/IAccount.java
* /opt/cola/permits/1256060467_1642635517.64/0/msal4j-1-2-0-sources-jar/com/microsoft/aad/msal4j/IPublicClientApplication.java
* /opt/cola/permits/1256060467_1642635517.64/0/msal4j-1-2-0-sources-jar/com/microsoft/aad/msal4j/DeviceCodeFlowParameters.java
* /opt/cola/permits/1256060467_1642635517.64/0/msal4j-1-2-0-sources-jar/com/microsoft/aad/msal4j/UserAssertion.java
* /opt/cola/permits/1256060467_1642635517.64/0/msal4j-1-2-0-sources-jar/com/microsoft/aad/msal4j/WSTrustRequest.java
* /opt/cola/permits/1256060467_1642635517.64/0/msal4j-1-2-0-sources-jar/com/microsoft/aad/msal4j/Event.java
* /opt/cola/permits/1256060467_1642635517.64/0/msal4j-1-2-0-sources-jar/com/microsoft/aad/msal4j/ClientApplicationBase.java
* /opt/cola/permits/1256060467_1642635517.64/0/msal4j-1-2-0-sources-jar/com/microsoft/aad/msal4j/XmsClientTelemetryInfo.java
// Copyright (c) Microsoft Corporation. All rights reserved.
//Licensed under the MIT License.
/**
 * Object containing parameters for silent requests. Can be used as parameter to
 * [PublicClientApplication#acquireTokenSilently(SilentParameters)] or to
 * [ConfidentialClientApplication#acquireTokenSilently(SilentParameters)]
 */

No license file was found, but licenses were detected in source scan.
Found in path(s):
* /opt/cola/permits/1256060467_1642635517.64/0/msal4j-1-2-0-sources-jar/com/microsoft/aad/msal4j/SilentParameters.java

No license file was found, but licenses were detected in source scan.

Signature-Version: 1.0
SHA-256-Digest-Manifest/Main-Attributes: ajLi/ZVDP0IygrDXf5SU5RtyHBrpw86qVUyK5hH00v6k=
SHA-256-Digest-Manifest: HflPRu4yLaiyRvKrE6+ovcNlZ93LLL4Qpi9DGlhQr1Q4=
Created-By: 1.8.0_112 (Oracle Corporation)

Name: com/microsoft/aad/msal4j/AccessTokenCacheEntity.java
SHA-256-Digest: Utbw5SgeDb66jfp4o4mA8Udk7j6TTbDRvf5bbgMcLGw=

Name: com/microsoft/aad/msal4j/DefaultHttpClient.java
SHA-256-Digest: 8Ec+3TckxexqbKnOCIXdCjC+rf0JjY0h3DRECykmQZQ=

Name: com/microsoft/aad/msal4j/ErrorResponse.java
SHA-256-Digest: QatufbpJiEJjmD0vo3KJWg+QiSPDC/g8VAXiwd8lcQo=

Name: com/microsoft/aad/msal4j/SafeDocumentBuilderFactory.java
SHA-256-Digest: eqIsFmV5K5Hh8HfPOAT9GmV0dntbDlIn83E5hOthMw=

Name: com/microsoft/aad/msal4j/ITokenCacheAccessAspect.java
SHA-256-Digest: to5LiSScJDdopa/+qMy6XLE8tEKUZOKd4c4WWPfCCDc=

Name: com/microsoft/aad/msal4j/ServiceBundle.java
SHA-256-Digest: IjaLjkIlyvGr/w2svKJrSoiyNjawBGYykZ5AV1wenEU=

Name: com/microsoft/aad/msal4j/IPublicClientApplication.java
SHA-256-Digest: 9aeW5+FZ2jndjAYOKdTrBRh7Vnh3Gp4fU90p95WQpec=

Name: com/microsoft/aad/msal4j/Authority.java
SHA-256-Digest: kCPgNYZIsXfIxX1rjZnDO7x09uWNxc/Om+IQFJABRM=

Name: com/microsoft/aad/msal4j/IHttpClient.java
SHA-256-Digest: 8o4cRoDtPQs73s5o4sOLNxcvyDTYDPuAw9PTGsdDDHA=

Name: com/microsoft/aad/msal4j/HttpMethod.java
SHA-256-Digest: 305SQ9Vh30BEqyX8XGeAKkh+q/hKivuPxlz56T4oQSI=

Name: com/microsoft/aad/msal4j/ITelemetryManager.java
SHA-256-Digest: PeLlb+2AW+yrs2a90MAdixymWTYYGhyMvyhEKLzx8WI=

Name: com/microsoft/aad/msal4j/IHttpClient.java
SHA-256-Digest: OFmwnNNJQxtT1zoZGdJxkb5ypJrtn1irv18Xktvy9sk=
SHA-256-Digest: h/vybuvAeo69LsHFIqXDSzH8hEu4U3Yr6dkc1cxhH5g=
Name: com/microsoft/aad/msal4j/UserDiscoveryRequest.java
SHA-256-Digest: r6d3ITzok+TlduMQFwhvzAAn66BZYR0+Z5gungpeZ/Y=
Name: com/microsoft/aad/msal4j/DeviceCodeAuthorizationGrant.java
SHA-256-Digest: Kk/No/YiaAyo5IE8W/gvpGO1CgkoIRup+FbVRVpnlaw=
Name: com/microsoft/aad/msal4j/IntegratedWindowsAuthenticationRequest.java
SHA-256-Digest: L32sOG4bfW8eE1CuBTf603pJ+HFlhQjS6amWxAz3GpSA=
Name: com/microsoft/aad/msal4j/CredentialTypeEnum.java
SHA-256-Digest: vH/QW7PZYoC+h04hxbOBQqSBSqVFcy9nHvRcCoB1m8=
Name: com/microsoft/aad/msal4j/Assertion.java
SHA-256-Digest: Wzpdcjxs+X8uXdfIQp1VJGfXMuqZ/QPLkvT3n0cJDg=
Name: com/microsoft/aad/msal4j/AccessToken.java
SHA-256-Digest: m8+hnTjL6zVstbN/PT7o/21f2MnxQta+VSCck6MY=
Name: com/microsoft/aad/msal4j/DeviceCodeAuthorizationGrant.java
SHA-256-Digest: MdTPJ1zlhjVSHsHPsDfrLq72twN6XxoWDMGcfKm90Y=
Name: com/microsoft/aad/msal4j/ITelemetry.java
SHA-256-Digest: WYn4mYe33w96PbbBN495eHwDoA1bex9HexSpDpGhqs=
Name: com/microsoft/aad/msal4j/AuthenticationResult.java
SHA-256-Digest: rAUHB0zXRGfT3pIOuUTE7Qn3Bi3JLPFPbnkFZUJS+oU=
Name: com/microsoft/aad/msal4j/IHttpClient.java
SHA-256-Digest: gl.8GV+TXIN+Xu4z3NfQrLmAc/j1R6uxUCzF2nefg=
Name: com/microsoft/aad/msal4j/SAML11BearerGrant.java
SHA-256-Digest: 4uLy+9gHSJmVGfoOqSRQInErKlfckh7Dq4kT5jWVgfI=
Name: com/microsoft/aad/msal4j/WSTrustVersion.java
SHA-256-Digest: 5adZtFehtYDHj/P+q+iPBuZSQw97X6V6lDAwLTHQTM=
Name: com/microsoft/aad/msal4j/BindingPolicy.java
SHA-256-Digest: gsmICCCXkF2ZyrMR6bjvKdnq42ByDRTZcAd1E30/VkM=
Name: com/microsoft/aad/msal4j/HttpEvent.java
SHA-256-Digest: rERF62H7nlua2j9BxXp60qQVzc9HLCe8CeOd1ARz3yQ=
Name: com/microsoft/aad/msal4j/HttpRequest.java
SHA-256-Digest: qLYmvkKF7kTZ1ZWqMCuoUvG0+i7tgYiGI0wq4qt4drE=
Name: com/microsoft/aad/msal4j/OAuthHttpRequest.java
SHA-256-Digest: +smyhEk27ThDeGKBeWmhmluoxNtQEElhSGgtuLH30dYA=

Name: com/microsoft/aad/msal4j/ITokenCacheAccessContext.java
SHA-256-Digest: pL+sisiAYoNL/ea9fGOuSjh4whEkOmtf2EVnwRSq0kVs=

Name: com/microsoft/aad/msal4j/TelemetryManager.java
SHA-256-Digest: WUkqTLav+Ar9AZMy1gBdhIFoHBAPBKBB+3dTtnMHh9I=

Name: com/microsoft/aad/msal4j/AcquireTokenByAuthorizationGrantSupplie r.java
SHA-256-Digest: QhMYGApocjh0dz/ua3FTNvuXTJls/JHopDU+EvO05IA=

Name: com/microsoft/aad/msal4j/ClientCredentialParameters.java
SHA-256-Digest: zogWmIoBYM6xHPRFzuMPJWw23nElfmyWOaHPmxTCeRk=

Name: com/microsoft/aad/msal4j/ClientApplicationBase.java
SHA-256-Digest: FfvK2Mczek0qwQu9dAH8z+JEJiX2jV9leD05ip6nHJo=

Name: com/microsoft/aad/msal4j/OAuthAuthorizationGrant.java
SHA-256-Digest: B9I8JpzN3aOj4j17I2Od4pTv3Z8bCrNYe+02mu6E0SY=

Name: com/microsoft/aad/msal4j/AuthenticationErrorCode.java
SHA-256-Digest: Of0a1MAetBeLS2Qhh2SILMeaBysokF6qoj1+SepWgtg=

Name: com/microsoft/aad/msal4j/Account.java
SHA-256-Digest: UJGtyyRltS/ci6SZhDaXYNjUeLeQ1zRw0RYgbQ4xye0=

Name: com/microsoft/aad/msal4j/ADFSAuthority.java
SHA-256-Digest: c+BUv6jVg+16xExZ78K6uFdDMtyQP/vjybSwOAh55lw=

Name: com/microsoft/aad/msal4j/RefreshTokenParameters.java
SHA-256-Digest: F7I4K5QVvRavYvlGX50p4MQOEKvkz/WwHmIwskdXoqo=
// Copyright (c) Microsoft Corporation. All rights reserved.
// Licensed under the MIT License.
/**
 * Interface to be implemented when configuring http client for [IPublicClientApplication] or
 * [IConfidentialClientApplication].
 */

Manifest-Version: 1.0
Archiver-Version: Plexus Archiver
Built-By: VssAdministrator
Created-By: Apache Maven
Build-Jdk: 1.8.0_102

Name: com/microsoft/aad/msal4j/AccessTokenCacheEntity.java
SHA-256-Digest: ZlMxHP9ex4HubDnywAUGin6BUPGxgkyosnL/Rh9SaY=

Name: com/microsoft/aad/msal4j/DefaultHttpClient.java
SHA-256-Digest: CNE/eBPZczyXYXysx+Fqcw/a/hV4xf8EtDni2VKuzk4=

Name: com/microsoft/aad/msal4j/ErrorResponse.java
SHA-256-Digest: qbgZOVSg8XodaMqre06gmE2XJFGFwepUwYPY5ZEtw4w=

Name: com/microsoft/aad/msal4j/SafeDocumentBuilderFactory.java
SHA-256-Digest: arnL2lMA8e1i3106Q1lKrU5uJXyXdfXl4RwBRCWt0=

Name: com/microsoft/aad/msal4j/ITokenCacheAccessAspect.java
SHA-256-Digest: Jf/LJNJ5AAZuOBnapr7fVRw0AMukiNms4YI6z60hX5c=

Name: com/microsoft/aad/msal4j/ServiceBundle.java
SHA-256-Digest: w0dVeDl9fYo9wcDgkZquXM8yq8dwjAuvaHQiH4tDUQ=

Name: com/microsoft/aad/msal4j/IPublicClientApplication.java
SHA-256-Digest: l9kyEXq08BACMLdPNNjEMd1qE5LCU5+N7S5ftTl/IBw=

Name: com/microsoft/aad/msal4j/Authority.java
SHA-256-Digest: rmK6acopYJZrPPVkJXdfOVOVngsSMm24O9tp3VnbAJEg=

Name: com/microsoft/aad/msal4j/IHttpResponse.java
SHA-256-Digest: HU5QDXwQ3EC+Mp/xKbqnWXXNaZrtQeA8Rq8iAzw7nVo=
Name: com/microsoft/aad/msal4j/HttpMethod.java
SHA-256-Digest: avXBdeLGDX8RJf0ITJoq/CyXwVU56f2W2uLCHXcwfFU=

Name: com/microsoft/aad/msal4j/ITelemetryManager.java
SHA-256-Digest: IJjdxtNhSjrH6NC7vCw7Pfa3a2oTWKneIzsc0wUpNk78=

Name: com/microsoft/aad/msal4j/IClientApplicationBase.java
SHA-256-Digest: rs+v3QXgW0p0i9I87jijyAuNv09iYxCXZ0B1fjZa3XU=

Name: com/microsoft/aad/msal4j/TokenResponse.java
SHA-256-Digest: iK2ZkwkWpRz1xAKuemYXgOrWnSW/VQtEhktGL0xVT7U=

Name: com/microsoft/aad/msal4j/AADAuthority.java
SHA-256-Digest: do9jo0Flh49JBvCgtE2yAJRlym9NenCGAzcLzvaP8G1Q=

Name: com/microsoft/aad/msal4j/HttpResponse.java
SHA-256-Digest: lDKwVWJL2QUEy/3ut0FOyixWkydWeklHa3PnS6SHkuw=

Name: com/microsoft/aad/msal4j/AuthenticationResultSupplier.java
SHA-256-Digest: 1yioayP9iLpA/quWrP9McHm3pLxI9KZ9XzD8u3zNM4=

Name: com/microsoft/aad/msal4j/AccountsSupplier.java
SHA-256-Digest: KYClP557mSxV6rYjQ/9WcpDddGHFpmvuH8aIZZg44ks=

Name: com/microsoft/aad/msal4j/InstanceDiscoveryMetadataEntry.java
SHA-256-Digest: 4Smh6H7okCpu/BebsdZin5dmfKxgz8/hXczgdQ5zEQw=

Name: com/microsoft/aad/msal4j/DeviceCodeFlowRequest.java
SHA-256-Digest: Huya4Sev7jmWhH01nv7s55ro5cRJohVvuQ/9Kcmu5Q=
Name: com/microsoft/aad/msal4j/SilentRequest.java
SHA-256-Digest: WdWfww4oxLcKnd/KQPTJ/DxcuUdJW+99Z18QKVhniIk=

Name: com/microsoft/aad/msal4j/ParameterValidationUtils.java
SHA-256-Digest: jPS79Nqhh/ZKe380tKBxG9KS2IHzl36Pmlq0POM200=

Name: com/microsoft/aad/msal4j/JwtHelper.java
SHA-256-Digest: MBCUlwPldq0JdpXbJatLvLUzmFYnVBqBw5iYuBphYs=

Name: com/microsoft/aad/msal4j/JsonHelper.java
SHA-256-Digest: ntw4+bJDLPZHuGRokEeeYxmBcnplDVIzEiDfYsNtkE=

Name: com/microsoft/aad/msal4j/StringHelper.java
SHA-256-Digest: 92kYZ/+pgqXVCj4qKhafyrY6XX8+GrMbrezzyW0S32ds=

Name: com/microsoft/aad/msal4j/IClientSecret.java
SHA-256-Digest: JigGXPNpy9sruxx9ZfiNKg8lKuH2RGqx8p+Kiilequbc=

Name: com/microsoft/aad/msal4j/IAuthenticationResult.java
SHA-256-Digest: hCdjuiapVnLR9BZDJgJnnM5Yxos8B62Dec0wDDpGAQfA=

Name: com/microsoft/aad/msal4j/RemoveAccountRunnable.java
SHA-256-Digest: ReKrWfeshK6lUIzu0CrDoruMkyk+iicu+Hq1tIS7Ks=

Name: com/microsoft/aad/msal4j/WSTrustResponse.java
SHA-256-Digest: iKpS4Lkh+jN58WEA75RqRjY3CZzTaqoCDTusMt6Q58=

Name: com/microsoft/aad/msal4j/AppMetadataCacheEntity.java
SHA-256-Digest: ri1IK5AGNOdwBlkgDuHZ1PL+Xv3FRVE0f6Bkihtc+Dk=

Name: com/microsoft/aad/msal4j/IntegratedWindowsAuthorizationGrant.java
SHA-256-Digest: nUcuZc+zE2T8kZJI0+NhBZOiJxr5iKH2eNyBK0ostM=

Name: com/microsoft/aad/msal4j/UserRequestExecutor.java
SHA-256-Digest: GUmA57KcSpwLhh4/gEzwpyrCsiO6osLqbHdkeUcs=

Name: com/microsoft/aad/msal4j/UserDiscoveryResponse.java
SHA-256-Digest: huec/GJCVe6cW7fuyr948/b6YbWT3eKqnW9sFlGbPWY=

Name: com/microsoft/aad/msal4j/TokenType.java
SHA-256-Digest: NsOmf2XgimzbZZ3fucTCooI2pJrJ/Kh2E78QfKZZdqQ=

Name: com/microsoft/aad/msal4j/AuthorizationCodeParameters.java
SHA-256-Digest: pWSKsz6GTKCXh0FmPtlbeUXVfLlhOPMFrrUpI4rzdpq=

Name: com/microsoft/aad/msal4j/IUserAssertion.java
SHA-256-Digest: HHcmDjKXXLTMNOd6kWXB7A3li/zGhvWKoQaLAPmuGRc=

Name: com/microsoft/aad/msal4j/Constants.java
SHA-256-Digest: i1O8SEMkUCUwTLbSM36y8Ag4K2J8VC1NBbN+7qYRh9Q=

Name: com/microsoft/aad/msal4j/UserNamePasswordRequest.java
SHA-256-Digest: +pSRXUhH79j+UslWxb0jWuOxEL8Qd/nIVx3tyMTWzM=

Name: com/microsoft/aad/msal4j/ClientCredentialFactory.java
SHA-256-Digest: puFfRIrXASK9mGFuRIZ41rmOccb6rAbPwb91KgCoo=

Name: com/microsoft/aad/msal4j/UserDiscoveryRequest.java
SHA-256-Digest: df45mjE00xJcBR4+MhXwYmJPvyuhOuDgyYnbZADutjw=

Name: com/microsoft/aad/msal4j/DeviceCodeAuthorizationGrant.java
SHA-256-Digest: schyri61qVoxOGb4vvgBCMqySCMjXe40s5yiEbdAoCI=

Name: com/microsoft/aad/msal4j/IntegratedWindowsAuthenticationRequest.java
SHA-256-Digest: 86q02C5WOCIuQ5gaKJjpcke5Lh/Erk5SG63IVkJBw=

Name: com/microsoft/aad/msal4j/CredentialTypeEnum.java
SHA-256-Digest: XTfuxPVj/UtKetbwmnpQkh5K9O1uNqszWX7aQhfIY=

Name: com/microsoft/aad/msal4j/UserAssertion.java
SHA-256-Digest: 9AcDeCV9eNIFatB0c1/yD4f6XTVWQfTCmwe/v/C+TJY=

Name: com/microsoft/aad/msal4j/AcquireTokenSilentSupplier.java
SHA-256-Digest: oj1A11xj38AJM9gRF6q26sfgjGz3IH0wHXqMEpxIAJQ=

Name: com/microsoft/aad/msal4j/IConfidentialClientApplication.java
SHA-256-Digest: 4ZB/TgYqJxRPJDZkDqGEZIEjzizIT+HGbC0lt+InKo=

Name: com/microsoft/aad/msal4j/ITelemetry.java
SHA-256-Digest: 2qFxtvOE0gO08L9uMSHvdG9ALQWHAoRFNZ4mkNnATE=

Name: com/microsoft/aad/msal4j/IHttpClient.java
SHA-256-Digest: 8bTJSZXNfjIP4e6lioZpYi3mpvjwyo26XAPgTofdydw=

Name: com/microsoft/aad/msal4j/IJwtClient.java
SHA-256-Digest: A5OdJE5u1B17KGMVNgUdI4k8AcJThpFynFXAaFRwpTs=

Name: com/microsoft/aad/msal4j/SAML11BearerGrant.java
SHA-256-Digest: BPyFPbST3D3hxGE+Oh9kJWaMvYc6SQDBqQSGbSH7E=

Name: com/microsoft/aad/msal4j/WSTrustVersion.java
SHA-256-Digest: Remxy6nzGRrwFvfcUlOtU9S/s49KciQpSZ+tFT2CY=
Name: com/microsoft/aad/msal4j/InteractionRequiredExceptionReason.java
SHA-256-Digest: lqnSI+GCf7vnaPIAZjen3/6gp8pT/+lEZkM0ZEBby3No=

Name: com/microsoft/aad/msal4j/MsalServiceExceptionFactory.java
SHA-256-Digest: p3Yash5nWOEmfZImhQW/qyLJ0+/B1xzYt9Gm/7f6ig=

Name: com/microsoft/aad/msal4j/IdTokenCacheEntity.java
SHA-256-Digest: g4/Ykp0qAeP4SkF8MT9JbVRY+gHxklwuW1sOAFqWMx0=

Name: com/microsoft/aad/msal4j/OAuthHttpRequest.java
SHA-256-Digest: vBar5Bpd3LVsle5CM7JBP89qv4yAFFxW/bigHAPZ8=

Name: com/microsoft/aad/msal4j/ITokenCacheAccessContext.java
SHA-256-Digest: g7zp5HppZVvQQGncqoPtmisiCahJZPct2dJX7TEs=

Name: com/microsoft/aad/msal4j/TelemetryManager.java
SHA-256-Digest: ybYTq/yx9GQPY2cT+mHKPmZ0P5MWHUJygNQ8dfXrZPg=

Name: com/microsoft/aad/msal4j/AcquireTokenByAuthorizationGrantSupplier.java
SHA-256-Digest: vGTT07bdFccfXPq0f8pi61NJNn8Nfi1/kkh/iRxY3M=

Name: com/microsoft/aad/msal4j/ClientCredentialParameters.java
SHA-256-Digest: niJ7yqjFOwrNI/b0J91kw/4m11B+TJhIyu52eDLkgk=

Name: com/microsoft/aad/msal4j/HttpHeaders.java
SHA-256-Digest: VkTC+HI/eRNpIIaNBxgF6z0Vdy/7rkiaZHO9UXpF0A=

Name: com/microsoft/aad/msal4j/ClientApplicationBase.java
SHA-256-Digest: 5s7Ww1j+7bK9gko4CwZqkJFpHkF/Tx6a3j0Srnvjnts=

Name: com/microsoft/aad/msal4j/OAuthAuthorizationGrant.java
SHA-256-Digest: y1MZ/zFH100cwog1piik3F6Y1KRmNet+WLyVjJNtY=

Name: com/microsoft/aad/msal4j/MsalException.java
SHA-256-Digest: RkCMTu5dOF7xeO1yLS0uZeOC1FBfWvuPLGFznBHyc=

Name: com/microsoft/aad/msal4j/ClientAuthenticationPost.java
SHA-256-Digest: xnAu4zdulCwas1jgDTHUyyLpHyHISxWvCy0VuRfxfpI=

Name: com/microsoft/aad/msal4j/MsalRequest.java
SHA-256-Digest: LzkzmClawyu4Bl3u5oq+d86GwKRgEEPgD/dt8UBY1U=

Name: com/microsoft/aad/msal4j/AuthenticationErrorCode.java
SHA-256-Digest: ZrU+AX3EqBJ9dyCROrNOdZYHmRZi8Kmmyn1YW4Gk=

Name: com/microsoft/aad/msal4j/Account.java
SHA-256-Digest: FCtAS5IH4yC174S2ULxKRP5WJZbBytLQrK7uBvrU7o=
Name: com/microsoft/aad/msal4j/ADFSAuthority.java
SHA-256-Digest: gEZou9BsOE0+RXr2PnkwOe+CudGlDNzsNBMcJ/wDRDE=
Name: com/microsoft/aad/msal4j/ITokenCache.java
SHA-256-Digest: cd8fP5DIfhFCiPm1LKjkbRjJL317Et6ASkTHucA4K8A=
Name: com/microsoft/aad/msal4j/RefreshTokenParameters.java
SHA-256-Digest: elleWSGmf1gZeYU5OQTgFlXympoM78+XTqu1X+ySj/k=
Name: com/microsoft/aad/msal4j/XmsClientTelemetryInfo.java
SHA-256-Digest: xSIgnwb01cn703DyTD30Y4X9C3bo6lKFKDhuBZsh8=
Name: com/microsoft/aad/msal4j/EventKey.java
SHA-256-Digest: jWVC4LSx/TyceAkKB6fw2T2Pp7U2GWiEmT15IJCNPg=
Name: com/microsoft/aad/msal4j/DefaultEvent.java
SHA-256-Digest: skJFf3PXiW7EmyJhPinoqr4TSIspre1sqAu7zoOY/Fg=
Name: com/microsoft/aad/msal4j/UserNamePasswordParameters.java
SHA-256-Digest: paz/R+V0MDsdQsL/t46Zx0psdDXSGev86W0Bk17RJ8=
Name: com/microsoft/aad/msal4j/PublicClientApplication.java
SHA-256-Digest: f6F8qOsEu1RdSaherMHAOU6k9ZXDAJCb1foyAZ/fj4BE=
Name: com/microsoft/aad/msal4j/DeviceCodeFlowParameters.java
SHA-256-Digest: U4o0CyoYo90T+L9W6lBY+LVTCk1k5wULCgO1mU7hlk=
Name: com/microsoft/aad/msal4j/AcquireTokenByDeviceCodeFlowSupplier.java
SHA-256-Digest: HQLxiqziGdrCMdBiDr0yetV5ykZgkDBwWc5wmRJhOQ=
Name: com/microsoft/aad/msal4j/MexParser.java
SHA-256-Digest: JIh3ZauFhfGx9riGH7M9tiHiEs3Kobbbk0hlR1siY4=
Name: com/microsoft/aad/msal4j/WSTrustRequest.java
SHA-256-Digest: ks/RKFA8nTF+RLtpGiTNsv2tA4wphjsePOLjzUywqo=
Name: com/microsoft/aad/msal4j/Event.java
SHA-256-Digest: 2uhNqQGg0CLLLQ0t3JOnMcau//SWA3ZTdwUqkQMM5W2U=
Name: com/microsoft/aad/msal4j/ClientSecret.java
SHA-256-Digest: RIOzAqzY938oFKRZIiu8pTymiv+dATb/oV98RIWJ3p0=
Name: com/microsoft/aad/msal4j/IClientCredential.java
SHA-256-Digest: maIptKrOv01PAqkv6SuOSjufYI+aJJW/7h56M1ujA=
Name: com/microsoft/aad/msal4j/MsalInteractionRequiredException.java
SHA-256-Digest: u+efsqhqwgu/k3FonI5S4IgR7hr6fsX1/2VMblCnpik=

Name: com/microsoft/aad/msal4j/RefreshTokenRequest.java
SHA-256-Digest: Yj89vVNFZtaX0naq2q6+s8PCsNL13eyn4IOCnbumQw90=

Name: com/microsoft/aad/msal4j/TelemetryConstants.java
SHA-256-Digest: 6oQPI3troJhK2nJOgxPSOmu05fhW7QSI+sgAIpwDJ4=

Name: com/microsoft/aad/msal4j/Credential.java
SHA-256-Digest: cQvwC0r+6lpWBOPofs3MGbZ3WVENCiv2jHUS0TKZDRA=

Name: com/microsoft/aad/msal4j/InstanceDiscoveryRequest.java
SHA-256-Digest: dkwWIxWMtgxY9C6WTcgorFaqNoi+l3fevC4gNeS0zE=

Name: com/microsoft/aad/msal4j/IClientCertificate.java
SHA-256-Digest: Jqh6OC0Sd0ytiiEJ/10dQI5WJCrvqwof+17dU14DIMiE=

Name: com/microsoft/aad/msal4j/TokenCacheAccessContext.java
SHA-256-Digest: U3hU4L8Pehh4o9o7tMN0f1hdj66CU/Cm/BTpvu3UqRI=

Name: com/microsoft/aad/msal4j/DeviceCode.java
SHA-256-Digest: 58hwc+jZ5hXNRygD2mEB1Pn53O/ZVADr7X+vyfDv0pc=

Name: com/microsoft/aad/msal4j/HttpClient.java
SHA-256-Digest: cEmiuUynQDPn6FheHJ+UDn1xMCHIe9RI7zmw1jORSs=
// Copyright (c) Microsoft Corporation. All rights reserved.
// Licensed under the MIT License.
/**
* Interface representing a confidential client application (Web App, Web API, Daemon App).
* Confidential client applications are trusted to safely store application secrets, and therefore
* can be used to acquire tokens in name of either the application or an user.
* For details see https://aka.ms/msal4jclientapplications
*/
/**
* Acquires tokens from the authority configured in the application, for the confidential client
* itself
* @param parameters instance of {@link ClientCredentialParameters}
* @return CompletableFuture containing an {@link IAuthenticationResult}
*/
/**
* Acquires an access token for this application (usually a Web API) from the authority configured
* in the application, in order to access another downstream protected Web API on behalf of a user
* using the On-Behalf-Of flow. This confidential client application was itself called with a token
* which will be provided in the {@link UserAssertion} to the {@link OnBehalfOfParameters}
* @param parameters instance of {@link OnBehalfOfParameters}
* @return CompletableFuture containing an {@link IAuthenticationResult}
*/

// Copyright (c) Microsoft Corporation. All rights reserved.
// Licensed under the MIT License.
*/
* Object containing parameters for refresh token request. Can be used as parameter to

* {@link PublicClientApplication#acquireToken(RefreshTokenParameters)} or to
* [@link ConfidentialClientApplication#acquireToken(RefreshTokenParameters)]
*/

Found in path(s):
* /opt/cola/permits/1256060467_1642635517.64/0/msal4j-1-2-0-sources-
jar/com/microsoft/aad/msal4j/RefreshTokenParameters.java

No license file was found, but licenses were detected in source scan.

// Copyright (c) Microsoft Corporation. All rights reserved.
// Licensed under the MIT License.
/**
* Class to be used to acquire tokens for confidential client applications (Web Apps, Web APIs,
* and daemon applications).
*/
/**
* Creates instance of Builder of ConfidentialClientApplication
* 
* @param clientId Client ID (Application ID) of the application as registered
* in the application registration portal (portal.azure.com)
* @param clientCredential The client credential to use for token acquisition.
* 
* @return instance of Builder of ConfidentialClientApplication
*/

Found in path(s):
* /opt/cola/permits/1256060467_1642635517.64/0/msal4j-1-2-0-sources-
jar/com/microsoft/aad/msal4j/ConfidentialClientApplication.java

1.107 quality-check 1.3
1.107.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*******************************************************************************
* Copyright 2013 André Rouél and Dominik Seichter
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
*   http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*******************************************************************************/
Found in path(s):
* /opt/cola/permits/1110944897_1606873389.6/0/quality-check-1-3-sources-1-jar/net/sf/qualitycheck/exception/IllegalPatternArgumentException.java
* /opt/cola/permits/1110944897_1606873389.6/0/quality-check-1-3-sources-1-jar/net/sf/qualitycheck/exception/IllegalRangeException.java
* /opt/cola/permits/1110944897_1606873389.6/0/quality-check-1-3-sources-1-jar/net/sf/qualitycheck/exception/IllegalNotGreaterOrEqualThanException.java
* /opt/cola/permits/1110944897_1606873389.6/0/quality-check-1-3-sources-1-jar/net/sf/qualitycheck/exception/IllegalNumberArgumentException.java
* /opt/cola/permits/1110944897_1606873389.6/0/quality-check-1-3-sources-1-jar/net/sf/qualitycheck/exception/IllegalNegativeArgumentException.java
* /opt/cola/permits/1110944897_1606873389.6/0/quality-check-1-3-sources-1-jar/net/sf/qualitycheck/exception/IllegalNullElementsException.java
* /opt/cola/permits/1110944897_1606873389.6/0/quality-check-1-3-sources-1-jar/net/sf/qualitycheck/exception/IllegalNullArgumentException.java
* /opt/cola/permits/1110944897_1606873389.6/0/quality-check-1-3-sources-1-jar/net/sf/qualitycheck/exception/RuntimeExceptionInstantiationException.java
* /opt/cola/permits/1110944897_1606873389.6/0/quality-check-1-3-sources-1-jar/net/sf/qualitycheck/exception/IllegalNotNullArgumentException.java
* /opt/cola/permits/1110944897_1606873389.6/0/quality-check-1-3-sources-1-jar/net/sf/qualitycheck/NumberInRange.java
* /opt/cola/permits/1110944897_1606873389.6/0/quality-check-1-3-sources-1-jar/net/sf/qualitycheck/exception/IllegalMissingAnnotationException.java
* /opt/cola/permits/1110944897_1606873389.6/0/quality-check-1-3-sources-1-jar/net/sf/qualitycheck/exception/IllegalPositionIndexException.java
* /opt/cola/permits/1110944897_1606873389.6/0/quality-check-1-3-sources-1-jar/net/sf/qualitycheck/exception/IllegalInstanceOfArgumentException.java
* /opt/cola/permits/1110944897_1606873389.6/0/quality-check-1-3-sources-1-jar/net/sf/qualitycheck/ArgumentsChecked.java
* /opt/cola/permits/1110944897_1606873389.6/0/quality-check-1-3-sources-1-jar/net/sf/qualitycheck/exception/IllegalNotLesserThanException.java
* /opt/cola/permits/1110944897_1606873389.6/0/quality-check-1-3-sources-1-jar/net/sf/qualitycheck/exception/IllegalPositiveArgumentException.java
* /opt/cola/permits/1110944897_1606873389.6/0/quality-check-1-3-sources-1-jar/net/sf/qualitycheck/exception/IllegalNaNArgumentException.java
* /opt/cola/permits/1110944897_1606873389.6/0/quality-check-1-3-sources-1-jar/net/sf/qualitycheck/exception/IllegalNumberRangeException.java
* /opt/cola/permits/1110944897_1606873389.6/0/quality-check-1-3-sources-1-jar/net/sf/qualitycheck/exception/IllegalArgumentNotContainedException.java
* /opt/cola/permits/1110944897_1606873389.6/0/quality-check-1-3-sources-1-jar/net/sf/qualitycheck/exception/IllegalEmptyArgumentException.java
* /opt/cola/permits/1110944897_1606873389.6/0/quality-check-1-3-sources-1-jar/net/sf/qualitycheck/exception/IllegalNumericArgumentException.java
* /opt/cola/permits/1110944897_1606873389.6/0/quality-check-1-3-sources-1-jar/net/sf/qualitycheck/exception/IllegalArgumentNotContainedException.java
1.108 freemarker 2.3.31
1.108.1 Available under license:

Licensed to the Apache Software Foundation (ASF) under one or more contributor license agreements. See the NOTICE file distributed with this work for additional information regarding copyright ownership. The ASF licenses this file to you under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTicular PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

========================================================================

The source code contains the following binaries, which were created at the Apache FreeMarker project, and hence are covered by the same license as the other source files of it:

src/main/misc/overloadedNumberRules/prices.ods
src/manual/en_US/docgen-originals/figures/overview.odg
src/manual/en_US/docgen-originals/figures/model2sketch_with_alpha.png
 TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License. You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of Your accepting any such warranty or additional liability.
APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

========================================================================= The Apache FreeMarker documentation contains the following binaries, which were created at the Apache FreeMarker project, and hence are covered by the same license as the other source files of it:

documentation/_html/docgen-resources/img/patterned-bg.png
documentation/_html/docgen-resources/img/xxe.png
documentation/_html/favicon.png
documentation/_html/figures/model2sketch.png
documentation/_html/figures/overview.png
documentation/_html/figures/tree.png
documentation/_html/logo.png

========================================================================= The documentation includes a selection of icons from various icon sets (fonts), stored together inside these files, which were generated with https://icomoon.io/app/:

documentation/_html/docgen-resources/fonts/icomoon.eot
The name, license, and attribution of each icon sets (fonts) used follows:

- The documentation includes icons from Entypo pictograms, version 2.0, by Daniel Bruce (http://www.entypo.com/, http://www.danielbruce.se/), licensed under Creative Commons Attribution-ShareAlike 3.0 (CC BY-SA 3.0) (http://creativecommons.org/licenses/by-sa/3.0/legalcode) and under SIL Open Font License 1.1 (http://scripts.sil.org/OFL).

- The documentation includes icons from Font Awesome by Dave Gandy (http://fontawesome.io), licensed under SIL Open Font License 1.1 (http://scripts.sil.org/OFL).

- The documentation includes icons from Material Design icons by Google (http://google.github.io/material-design-icons/), licensed under Creative Common Attribution 4.0 International License (CC-BY 4.0) (https://creativecommons.org/licenses/by/4.0/).

=========================================================================

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.
"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.
You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.109 uadetector-resources 2014.10
1.109.1 Available under license:

****** Creative Commons Legal Code ******

http://creativecommons.org/licenses/by/3.0/legalcode

****** Attribution 3.0 Unported ******

CREATIVE COMMONS CORPORATION IS NOT A LAW FIRM AND DOES NOT PROVIDE LEGAL SERVICES. DISTRIBUTION OF THIS LICENSE DOES NOT CREATE AN ATTORNEY-CLIENT RELATIONSHIP. CREATIVE COMMONS PROVIDES THIS INFORMATION ON AN "AS-IS" BASIS. CREATIVE COMMONS MAKES NO WARRANTIES REGARDING THE INFORMATION PROVIDED, AND DISCLAIMS LIABILITY FOR
DAMAGES RESULTING FROM ITS USE.

**** License ****

THE WORK (AS DEFINED BELOW) IS PROVIDED UNDER THE TERMS OF THIS CREATIVE COMMONS PUBLIC LICENSE ("CCPL" OR "LICENSE"). THE WORK IS PROTECTED BY COPYRIGHT AND/OR OTHER APPLICABLE LAW. ANY USE OF THE WORK OTHER THAN AS AUTHORIZED UNDER THIS LICENSE OR COPYRIGHT LAW IS PROHIBITED. 

BY EXERCISING ANY RIGHTS TO THE WORK PROVIDED HERE, YOU ACCEPT AND AGREE TO BE BOUND BY THE TERMS OF THIS LICENSE. TO THE EXTENT THIS LICENSE MAY BE CONSIDERED TO BE A CONTRACT, THE LICENSOR GRANTS YOU THE RIGHTS CONTAINED HERE IN CONSIDERATION OF YOUR ACCEPTANCE OF SUCH TERMS AND CONDITIONS.

1. Definitions
   a. "Adaptation" means a work based upon the Work, or upon the Work and other pre-existing works, such as a translation, adaptation, derivative work, arrangement of music or other alterations of a literary or artistic work, or phonogram or performance and includes cinematographic adaptations or any other form in which the Work may be recast, transformed, or adapted including in any form recognizably derived from the original, except that a work that constitutes a Collection will not be considered an Adaptation for the purpose of this License. For the avoidance of doubt, where the Work is a musical work, performance or phonogram, the synchronization of the Work in timed-relation with a moving image ("synching") will be considered an Adaptation for the purpose of this License.
   b. "Collection" means a collection of literary or artistic works, such as encyclopedias and anthologies, or performances, phonograms or broadcasts, or other works or subject matter other than works listed in Section 1(f) below, which, by reason of the selection and arrangement of their contents, constitute intellectual creations, in which the Work is included in its entirety in unmodified form along with one or more other contributions, each constituting separate and independent works in themselves, which together are assembled into a collective whole. A work that constitutes a Collection will not be considered an Adaptation (as defined above) for the purposes of this License.
   c. "Distribute" means to make available to the public the original and copies of the Work or Adaptation, as appropriate, through sale or other transfer of ownership.
   d. "Licensor" means the individual, individuals, entity or entities that offer(s) the Work under the terms of this License.
   e. "Original Author" means, in the case of a literary or artistic work, the individual, individuals, entity or entities who created the Work or if no individual or entity can be identified, the publisher; and in addition (i) in the case of a performance the actors, singers, musicians, dancers, and other persons who act, sing, deliver, declaim, play in, interpret or otherwise perform literary or artistic works or expressions of folklore; (ii) in the case of a phonogram the producer being the person or legal
entity who first fixes the sounds of a performance or other sounds; and, (iii) in the case of broadcasts, the organization that transmits the broadcast.

f. "Work" means the literary and/or artistic work offered under the terms of this License including without limitation any production in the literary, scientific and artistic domain, whatever may be the mode or form of its expression including digital form, such as a book, pamphlet and other writing; a lecture, address, sermon or other work of the same nature; a dramatic or dramatico-musical work; a choreographic work or entertainment in dumb show; a musical composition with or without words; a cinematographic work to which are assimilated works expressed by a process analogous to cinematography; a work of drawing, painting, architecture, sculpture, engraving or lithography; a photographic work to which are assimilated works expressed by a process analogous to photography; a work of applied art; an illustration, map, plan, sketch or three-dimensional work relative to geography, topography, architecture or science; a performance; a broadcast; a phonogram; a compilation of data to the extent it is protected as a copyrightable work; or a work performed by a variety or circus performer to the extent it is not otherwise considered a literary or artistic work.

g. "You" means an individual or entity exercising rights under this License who has not previously violated the terms of this License with respect to the Work, or who has received express permission from the Licensor to exercise rights under this License despite a previous violation.

h. "Publicly Perform" means to perform public recitations of the Work and to communicate to the public those public recitations, by any means or process, including by wire or wireless means or public digital performances; to make available to the public Works in such a way that members of the public may access these Works from a place and at a place individually chosen by them; to perform the Work to the public by any means or process and the communication to the public of the performances of the Work, including by public digital performance; to broadcast and rebroadcast the Work by any means including signs, sounds or images.

i. "Reproduce" means to make copies of the Work by any means including without limitation by sound or visual recordings and the right of fixation and reproducing fixations of the Work, including storage of a protected performance or phonogram in digital form or other electronic medium.

2. Fair Dealing Rights. Nothing in this License is intended to reduce, limit, or restrict any uses free from copyright or rights arising from limitations or exceptions that are provided for in connection with the copyright protection under copyright law or other applicable laws.

3. License Grant. Subject to the terms and conditions of this License, Licensor hereby grants You a worldwide, royalty-free, non-exclusive, perpetual (for the
duration of the applicable copyright) license to exercise the rights in the Work as stated below:

a. to Reproduce the Work, to incorporate the Work into one or more Collections, and to Reproduce the Work as incorporated in the Collections;
b. to create and Reproduce Adaptations provided that any such Adaptation, including any translation in any medium, takes reasonable steps to clearly label, demarcate or otherwise identify that changes were made to the original Work. For example, a translation could be marked "The original work was translated from English to Spanish," or a modification could indicate "The original work has been modified."

c. to Distribute and Publicly Perform the Work including as incorporated in Collections; and,
d. to Distribute and Publicly Perform Adaptations.
e. For the avoidance of doubt:
   i. Non-waivable Compulsory License Schemes. In those jurisdictions in which the right to collect royalties through any statutory or compulsory licensing scheme cannot be waived, the Licensor reserves the exclusive right to collect such royalties for any exercise by You of the rights granted under this License;
   ii. Waivable Compulsory License Schemes. In those jurisdictions in which the right to collect royalties through any statutory or compulsory licensing scheme can be waived, the Licensor waives the exclusive right to collect such royalties for any exercise by You of the rights granted under this License; and,
   iii. Voluntary License Schemes. The Licensor waives the right to collect royalties, whether individually or, in the event that the Licensor is a member of a collecting society that administers voluntary licensing schemes, via that society, from any exercise by You of the rights granted under this License.

The above rights may be exercised in all media and formats whether now known or hereafter devised. The above rights include the right to make such modifications as are technically necessary to exercise the rights in other media and formats. Subject to Section 8(f), all rights not expressly granted by Licensor are hereby reserved.

4. Restrictions. The license granted in Section 3 above is expressly made subject to and limited by the following restrictions:

a. You may Distribute or Publicly Perform the Work only under the terms of this License. You must include a copy of, or the Uniform Resource Identifier (URI) for, this License with every copy of the Work You Distribute or Publicly Perform. You may not offer or impose any terms on the Work that restrict the terms of this License or the ability of the recipient of the Work to exercise the rights granted to that recipient under the terms of the License. You may not sublicense the Work. You must keep intact all notices that refer to this License and to the disclaimer of warranties with every copy of the Work You Distribute or Publicly
Perform. When You Distribute or Publicly Perform the Work, You may not impose any effective technological measures on the Work that restrict the ability of a recipient of the Work from You to exercise the rights granted to that recipient under the terms of the License. This Section 4(a) applies to the Work as incorporated in a Collection, but this does not require the Collection apart from the Work itself to be made subject to the terms of this License. If You create a Collection, upon notice from any Licensor You must, to the extent practicable, remove from the Collection any credit as required by Section 4(b), as requested. If You create an Adaptation, upon notice from any Licensor You must, to the extent practicable, remove from the Adaptation any credit as required by Section 4(b), as requested.

b. If You Distribute, or Publicly Perform the Work or any Adaptations or Collections, You must, unless a request has been made pursuant to Section 4(a), keep intact all copyright notices for the Work and provide, reasonable to the medium or means You are utilizing: (i) the name of the Original Author (or pseudonym, if applicable) if supplied, and/or if the Original Author and/or Licensor designate another party or parties (e.g., a sponsor institute, publishing entity, journal) for attribution ("Attribution Parties") in Licensor's copyright notice, terms of service or by other reasonable means, the name of such party or parties; (ii) the title of the Work if supplied; (iii) to the extent reasonably practicable, the URI, if any, that Licensor specifies to be associated with the Work, unless such URI does not refer to the copyright notice or licensing information for the Work; and (iv), consistent with Section 3(b), in the case of an Adaptation, a credit identifying the use of the Work in the Adaptation (e.g., "French translation of the Work by Original Author," or "Screenplay based on original Work by Original Author"). The credit required by this Section 4(b) may be implemented in any reasonable manner; provided, however, that in the case of a Adaptation or Collection, at a minimum such credit will appear, if a credit for all contributing authors of the Adaptation or Collection appears, then as part of these credits and in a manner at least as prominent as the credits for the other contributing authors. For the avoidance of doubt, You may only use the credit required by this Section for the purpose of attribution in the manner set out above and, by exercising Your rights under this License, You may not implicitly or explicitly assert or imply any connection with, sponsorship or endorsement by the Original Author, Licensor and/or Attribution Parties, as appropriate, of You or Your use of the Work, without the separate, express prior written permission of the Original Author, Licensor and/or Attribution Parties.

c. Except as otherwise agreed in writing by the Licensor or as may be otherwise permitted by applicable law, if You Reproduce, Distribute or Publicly Perform the Work either by itself or as part of any Adaptations or Collections, You must not distort, mutilate, modify or take other derogatory action in relation to the Work which would be prejudicial to the Original Author's honor or reputation. Licensor agrees that in those jurisdictions (e.g. Japan), in which any exercise of the right granted in
Section 3(b) of this License (the right to make Adaptations) would be deemed to be a distortion, mutilation, modification or other derogatory action prejudicial to the Original Author's honor and reputation, the Licensor will waive or not assert, as appropriate, this Section, to the fullest extent permitted by the applicable national law, to enable You to reasonably exercise Your right under Section 3(b) of this License (right to make Adaptations) but not otherwise.

5. Representations, Warranties and Disclaimer

UNLESS OTHERWISE MUTUALLY AGREED TO BY THE PARTIES IN WRITING, LICENSOR OFFERS THE WORK AS-IS AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING THE WORK, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT, OR THE ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY, OR THE PRESENCE OF ABSENCE OF ERRORS, WHETHER OR NOT DISCOVERABLE. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO SUCH EXCLUSION MAY NOT APPLY TO YOU.

6. Limitation on Liability. EXCEPT TO THE EXTENT REQUIRED BY APPLICABLE LAW, IN NO EVENT WILL LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY FOR ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES ARISING OUT OF THIS LICENSE OR THE USE OF THE WORK, EVEN IF LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. Termination

a. This License and the rights granted hereunder will terminate automatically upon any breach by You of the terms of this License. Individuals or entities who have received Adaptations or Collections from You under this License, however, will not have their licenses terminated provided such individuals or entities remain in full compliance with those licenses. Sections 1, 2, 5, 6, 7, and 8 will survive any termination of this License.

b. Subject to the above terms and conditions, the license granted here is perpetual (for the duration of the applicable copyright in the Work). Notwithstanding the above, Licensor reserves the right to release the Work under different license terms or to stop distributing the Work at any time; provided, however that any such election will not serve to withdraw this License (or any other license that has been, or is required to be, granted under the terms of this License), and this License will continue in full force and effect unless terminated as stated above.

8. Miscellaneous

a. Each time You Distribute or Publicly Perform the Work or a Collection, the Licensor offers to the recipient a license to the Work on the same terms and conditions as the license granted to You under this License.
b. Each time You Distribute or Publicly Perform an Adaptation, Licensor offers to the recipient a license to the original Work on the same terms and conditions as the license granted to You under this License.
c. If any provision of this License is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this License, and without further action by the parties to this agreement, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.
d. No term or provision of this License shall be deemed waived and no breach consented to unless such waiver or consent shall be in writing and signed by the party to be charged with such waiver or consent.
e. This License constitutes the entire agreement between the parties with respect to the Work licensed here. There are no understandings, agreements or representations with respect to the Work not specified here. Licensor shall not be bound by any additional provisions that may appear in any communication from You. This License may not be modified without the mutual written agreement of the Licensor and You.
f. The rights granted under, and the subject matter referenced, in this License were drafted utilizing the terminology of the Berne Convention for the Protection of Literary and Artistic Works (as amended on September 28, 1979), the Rome Convention of 1961, the WIPO Copyright Treaty of 1996, the WIPO Performances and Phonograms Treaty of 1996 and the Universal Copyright Convention (as revised on July 24, 1971). These rights and subject matter take effect in the relevant jurisdiction in which the License terms are sought to be enforced according to the corresponding provisions of the implementation of those treaty provisions in the applicable national law. If the standard suite of rights granted under applicable copyright law includes additional rights not granted under this License, such additional rights are deemed to be included in the License; this License is not intended to restrict the license of any rights under applicable law.

**** Creative Commons Notice ****

Creative Commons is not a party to this License, and makes no warranty whatsoever in connection with the Work. Creative Commons will not be liable to You or any party on any legal theory for any damages whatsoever, including without limitation any general, special, incidental or consequential damages arising in connection to this license. Notwithstanding the foregoing two (2) sentences, if Creative Commons has expressly identified itself as the Licensor hereunder, it shall have all rights and obligations of Licensor. Except for the limited purpose of indicating to the public that the Work is licensed under the CCPL, Creative Commons does not authorize the use by either party of the trademark "Creative Commons" or any
related trademark or logo of Creative Commons without the prior written consent of Creative Commons. Any permitted use will be in compliance with Creative Commons' then-current trademark usage guidelines, as may be published on its website or otherwise made available upon request from time to time. For the avoidance of doubt, this trademark restriction does not form part of this License. Creative Commons may be contacted at http://creativecommons.org/.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright 2012 Andr Roul

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

The UADetector Project

Please visit the UADetector web site for more information:

* http://uadetector.sourceforge.net/

Copyright 2012 Andr Roul

Andr Roul licenses this product to you under the Apache License, version 2.0
(the "License"); you may not use this product except in compliance with the
License. You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed
under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR
CONDITIONS OF ANY KIND, either express or implied. See the License for the
specific language governing permissions and limitations under the License.

Also, please refer to each LICENSE.<component>.txt file, which is located in
the same directory as this file, for the license terms of the components that
this product depends on.

This product contains a version of Jaroslav Mallat's UAS Data, which can be
obtained at:

* LICENSE:
  * LICENSE.uas.txt (CC BY 3.0)
1.110 jackson-jaxrs 1.9.2

1.111 commons-codec 1.15

1.111.1 Available under license:

Apache Commons Codec
Copyright 2002-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (https://www.apache.org/).

src/test/org/apache/commons/codec/language/DoubleMetaphoneTest.java
contains test data from http://aspell.net/test/orig/batch0.tab.
Copyright (C) 2002 Kevin Atkinson (kevina@gnu.org)

===============================================================================

The content of package org.apache.commons.codec.language.bm has been translated
from the original php source code available at http://stevemorse.org/phoneticinfo.htm
with permission from the original authors.
Original source copyright:
Copyright (c) 2008 Alexander Beider & Stephen P. Morse.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.112 metrics 4.1.26

1.112.1 Available under license :

No license file was found, but licenses were detected in source scan.

Manifest-Version: 1.0
Automatic-Module-Name: com.codahale.metrics
Build-Jdk: 1.8.0_302
Built-By: runner
Bundle-Description: Metrics is a Java library which gives you unparalleled insight into what your code does in production. Metrics provides a powerful toolkit of ways to measure the behavior of critical components in your production environment.
Bundle-License: https://www.apache.org/licenses/LICENSE-2.0.html
Bundle-ManifestVersion: 2
Bundle-Name: Metrics Core
Bundle-SymbolicName: io.dropwizard.metrics.core
Bundle-Version: 4.1.26
Created-By: Apache Maven Bundle Plugin
Export-Package: com.codahale.metrics;uses:="org.slf4j";version="4.1.26"

Implementation-Title: Metrics Core
Implementation-URL: https://metrics.dropwizard.io/metrics-core
Implementation-Vendor-Id: io.dropwizard.metrics
Implementation-Version: 4.1.26
Import-Package: org.slf4j;version="[1.6.0,2.0.0)"
Private-Package: com.codahale.metrics
Require-Capability: osgi.ee;filter:="(&(osgi.ee=JavaSE)(version=1.8))"
Tool: Bnd-5.1.1.202006162103

Found in path(s):
* /opt/cola/permits/1221016569_1645091946.67/0/metrics-core-4-1-26-jar/META-INF/MANIFEST.MF

1.113 commons-compress 1.18

1.113.1 Available under license:
Apache License

Version 2.0, January 2004

http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.
"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following

---

Open Source Used In security_automation identity-guest-cs 627
You must give any other recipients of the Work or Derivative Works a copy of this License; and
You must cause any modified files to carry prominent notices stating that You changed the files; and
You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark,
and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of
the Derivative Works; and
If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute
must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices
that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text
file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the
Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices
normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the
License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an
addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license
terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works
as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions
stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for
inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any
additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of
any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product
names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and
reproducing the content of the NOTICE file.

7.Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work
(and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR
CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or
conditions of TITLE, NONINFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR
PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and
assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or
otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing,
shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or
consequential damages of any character arising as a result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or
any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of
such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may
choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or
rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf
and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: HOW TO APPLY THE APACHE LICENSE TO YOUR WORK
To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.114 joda-time 2.10.6
1.114.1 Available under license :

============================================================================== NOTICE file corresponding to section 4d of the Apache License Version 2.0 =
==============================================================================
This product includes software developed by
Joda.org (https://www.joda.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.
"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "["] replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
1.115 hibernate-validator 5.0.2.Final

1.115.1 Available under license:

Found license 'GNU Lesser General Public License' in * This is free software; you can redistribute it and/or modify it * under the terms of the GNU Lesser General Public License as * published by the Free Software Foundation; either version 2.1 of * the License, or (at your option) any later version. * This software is distributed in the hope that it will be useful, * but WITHOUT ANY WARRANTY; without even the implied warranty of * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU * Lesser General Public License for more details. * You should have received a copy of the GNU Lesser General Public * Expert Group and released to the public domain, as explained at'

1.116 javax-activation 1.2.0

1.116.1 Available under license:

COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.1

1. Definitions.

1.1. "Contributor" means each individual or entity that creates or contributes to the creation of Modifications.

1.2. "Contributor Version" means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. "Covered Software" means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. "Executable" means the Covered Software in any form other than Source Code.

1.5. "Initial Developer" means the individual or entity that first makes Original Software available under this License.

1.6. "Larger Work" means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. "License" means this document.

1.8. "Licensable" means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. "Modifications" means the Source Code and Executable form of
any of the following:

A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

B. Any new file that contains any part of the Original Software or previous Modification; or

C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. "Original Software" means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. "Patent Claims" means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. "Source Code" means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. "You" (or "Your") means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, "You" includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and
(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3)
under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients' rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.
3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.


You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.

Oracle is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.
4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as "Participant") alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.
6.3. If You assert a patent infringement claim against Participant alleging that the Participant Software directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

6.4. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a "commercial item," as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of "commercial computer software" (as that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and "commercial computer software documentation" as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.
This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

------------------------------------------------------------------------

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)

The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will
individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is, to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.
c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to
distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues),
conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the
Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.
Copyright (C) <year> <name of author>
This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1335 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

    signature of Ty Coon, 1 April 1989
    Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

#
Certain source files distributed by Oracle America, Inc. and/or its affiliates are subject to the following clarification and special exception to the GPLv2, based on the GNU Project exception for its Classpath libraries, known as the GNU Classpath Exception, but only where Oracle has expressly included in the particular source file's header the words "Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the LICENSE file that accompanied this code."

You should also note that Oracle includes multiple, independent programs in this software package. Some of those programs are provided under licenses deemed incompatible with the GPLv2 by the Free Software Foundation and others. For example, the package includes programs licensed under the Apache License, Version 2.0. Such programs are licensed to you under their original licenses.

Oracle facilitates your further distribution of this package by adding the Classpath Exception to the necessary parts of its GPLv2 code, which permits you to use that code in combination with other independent modules not licensed under the GPLv2. However, note that this would not permit you to commingle code under an incompatible license with Oracle's GPLv2 licensed code by, for example, cutting and pasting such code into a file also containing Oracle's GPLv2 licensed code and then distributing the result. Additionally, if you were to remove the Classpath Exception from any of the files to which it applies and distribute the result, you would likely be required to license some or all of the other code in that distribution under the GPLv2 as well, and since the GPLv2 is incompatible with the license terms of some items included in the distribution by Oracle, removing the Classpath Exception could therefore effectively compromise your ability to further distribute the package.

Proceed with caution and we recommend that you obtain the advice of a lawyer skilled in open source matters before removing the Classpath Exception or making modifications to this package which may subsequently be redistributed and/or involve the use of third party software.

CLASSPATH EXCEPTION
Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under
terms of your choice, provided that you also meet, for each linked
independent module, the terms and conditions of the license of that
module. An independent module is a module which is not derived from or
based on this library. If you modify this library, you may extend this
exception to your version of the library, but you are not obligated to
do so. If you do not wish to do so, delete this exception statement
from your version.

<p>
LIMITED LICENSE GRANTS
</p>

1. License for Evaluation Purposes. Specification Lead hereby grants
you a fully-paid, non-exclusive, non-transferable, worldwide, limited
license (without the right to sublicense), under Specification Lead’s
applicable intellectual property rights to view, download, use and
reproduce the Specification only for the purpose of internal
evaluation. This includes (i) developing applications intended to run
on an implementation of the Specification, provided that such
applications do not themselves implement any portion(s) of the
Specification, and (ii) discussing the Specification with any third
party; and (iii) excerpting brief portions of the Specification in oral
or written communications which discuss the Specification provided that
such excerpts do not in the aggregate constitute a significant portion
of the Specification.

</p>

2. License for the Distribution of Compliant Implementations.
Specification Lead also grants you a perpetual, non-exclusive,
non-transferable, worldwide, fully paid-up, royalty free, limited
license (without the right to sublicense) under any applicable
copyrights or, subject to the provisions of subsection 4 below, patent rights it may have covering the Specification to create and/or distribute an Independent Implementation of the Specification that: (a) fully implements the Specification including all its required interfaces and functionality; (b) does not modify, subset, superset or otherwise extend the Licensor Name Space, or include any public or protected packages, classes, Java interfaces, fields or methods within the Licensor Name Space other than those required/authorized by the Specification or Specifications being implemented; and (c) passes the Technology Compatibility Kit (including satisfying the requirements of the applicable TCK Users Guide) for such Specification ("Compliant Implementation"). In addition, the foregoing license is expressly conditioned on your not acting outside its scope. No license is granted hereunder for any other purpose (including, for example, modifying the Specification, other than to the extent of your fair use rights, or distributing the Specification to third parties). Also, no right, title, or interest in or to any trademarks, service marks, or trade names of Specification Lead or Specification Lead's licensors is granted hereunder. Java, and Java-related logos, marks and names are trademarks or registered trademarks of Oracle America, Inc. in the U.S. and other countries.

3. Pass-through Conditions. You need not include limitations (a)-(c) from the previous paragraph or any other particular "pass through" requirements in any license You grant concerning the use of your Independent Implementation or products derived from it. However, except with respect to Independent Implementations (and products derived from them) that satisfy limitations (a)-(c) from the previous paragraph, You may neither: (a) grant or otherwise pass through to your licensees any licenses under Specification Lead’s applicable intellectual property rights; nor (b) authorize your licensees to make any claims concerning their implementation’s compliance with the Specification in question.


a. With respect to any patent claims covered by the license granted under subparagraph 2 above that would be infringed by all technically feasible implementations of the Specification, such license is conditioned upon your offering on fair, reasonable and non-discriminatory terms, to any party seeking it from You, a perpetual, non-exclusive, non-transferable, worldwide license under Your patent rights which are or would be infringed by all technically feasible implementations of the Specification to develop, distribute and use a Compliant Implementation.
b. With respect to any patent claims owned by Specification Lead and covered by the license granted under subparagraph 2, whether or not their infringement can be avoided in a technically feasible manner when implementing the Specification, such license shall terminate with respect to such claims if You initiate a claim against Specification Lead that it has, in the course of performing its responsibilities as the Specification Lead, induced any other entity to infringe Your patent rights.

<br/>

c. Also with respect to any patent claims owned by Specification Lead and covered by the license granted under subparagraph 2 above, where the infringement of such claims can be avoided in a technically feasible manner when implementing the Specification such license, with respect to such claims, shall terminate if You initiate a claim against Specification Lead that its making, having made, using, offering to sell, selling or importing a Compliant Implementation infringes Your patent rights.

</p>

</p>

5. Definitions. For the purposes of this Agreement: "Independent Implementation" shall mean an implementation of the Specification that neither derives from any of Specification Lead's source code or binary code materials nor, except with an appropriate and separate license from Specification Lead, includes any of Specification Lead's source code or binary code materials; "Licensor Name Space" shall mean the public class or interface declarations whose names begin with "java", "javax", "com.oracle", "com.sun" or their equivalents in any subsequent naming convention adopted by Oracle America, Inc. through the Java Community Process, or any recognized successors or replacements thereof; and "Technology Compatibility Kit" or "TCK" shall mean the test suite and accompanying TCK User's Guide provided by Specification Lead which corresponds to the Specification and that was available either (i) from Specification Lead 120 days before the first release of Your Independent Implementation that allows its use for commercial purposes, or (ii) more recently than 120 days from such release but against which You elect to test Your implementation of the Specification.

</p>

</p>

This Agreement will terminate immediately without notice from Specification Lead if you breach the Agreement or act outside the scope of the licenses granted above.

</p>

</p>

DISCLAIMER OF WARRANTIES

<br/>

THE SPECIFICATION IS PROVIDED "AS IS". SPECIFICATION LEAD MAKES NO REPRESENTATIONS OR WARRANTIES, EITHER EXPRESS OR IMPLIED, INCLUDING BUT
NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NON-INFRINGEMENT (INCLUDING AS A CONSEQUENCE OF ANY PRACTICE OR IMPLEMENTATION OF THE SPECIFICATION), OR THAT THE CONTENTS OF THE SPECIFICATION ARE SUITABLE FOR ANY PURPOSE. This document does not represent any commitment to release or implement any portion of the Specification in any product. In addition, the Specification could include technical inaccuracies or typographical errors.

LIMITATION OF LIABILITY

TO THE EXTENT NOT PROHIBITED BY LAW, IN NO EVENT WILL SPECIFICATION LEAD OR ITS LICENSORS BE LIABLE FOR ANY DAMAGES, INCLUDING WITHOUT LIMITATION, LOST REVENUE, PROFITS OR DATA, OR FOR SPECIAL, INDIRECT, CONSEQUENTIAL, INCIDENTAL OR PUNITIVE DAMAGES, HOWEVER CAUSED AND REGARDLESS OF THE THEORY OF LIABILITY, ARISING OUT OF OR RELATED IN ANY WAY TO YOUR HAVING, IMPLEMENTING OR OTHERWISE USING THE SPECIFICATION, EVEN IF SPECIFICATION LEAD AND/OR ITS LICENSORS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. You will indemnify, hold harmless, and defend Specification Lead and its licensors from any claims arising or resulting from: (i) your use of the Specification; (ii) the use or distribution of your Java application, applet and/or implementation; and/or (iii) any claims that later versions or releases of any Specification furnished to you are incompatible with the Specification provided to you under this license.

RESTRICTED RIGHTS LEGEND

U.S. Government: If this Specification is being acquired by or on behalf of the U.S. Government or by a U.S. Government prime contractor or subcontractor (at any tier), then the Government's rights in the Software and accompanying documentation shall be only as set forth in this license; this is in accordance with 48 C.F.R. 227.7201 through 227.7202-4 (for Department of Defense (DoD) acquisitions) and with 48 C.F.R. 2.101 and 12.212 (for non-DoD acquisitions).

REPORT

If you provide Specification Lead with any comments or suggestions concerning the Specification ("Feedback"), you hereby: (i) agree that such Feedback is provided on a non-proprietary and non-confidential basis, and (ii) grant Specification Lead a perpetual, non-exclusive, worldwide, fully paid-up, irrevocable license, with the right to sublicense through multiple levels of sublicensees, to incorporate, disclose, and use without limitation the Feedback for any purpose.
GENERAL TERMS

Any action related to this Agreement will be governed by California law and controlling U.S. federal law. The U.N. Convention for the International Sale of Goods and the choice of law rules of any jurisdiction will not apply.

The Specification is subject to U.S. export control laws and may be subject to export or import regulations in other countries. Licensee agrees to comply strictly with all such laws and regulations and acknowledges that it has the responsibility to obtain such licenses to export, re-export or import as may be required after delivery to Licensee.

This Agreement is the parties' entire agreement relating to its subject matter. It supersedes all prior or contemporaneous oral or written communications, proposals, conditions, representations and warranties and prevails over any conflicting or additional terms of any quote, order, acknowledgment, or other communication between the parties relating to its subject matter during the term of this Agreement. No modification to this Agreement will be binding, unless in writing and signed by an authorized representative of each party.

1.117 swagger-annotations 1.5.21

1.117.1 Available under license:

/**
 * Copyright 2016 SmartBear Software
 * <p>
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * <p>
 * http://www.apache.org/licenses/LICENSE-2.0
 * <p>
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
package io.swagger.annotations;

import java.lang.annotation.ElementType;
import java.lang.annotation.Retention;
import java.lang.annotation.RetentionPolicy;
import java.lang.annotation.Target;

/**
 * License metadata available within the info section of a Swagger definition, see
 * https://github.com/OAI/OpenAPI-Specification/blob/master/versions/2.0.md#licenseObject
 * @since 1.5.0
 */

@Target(ElementType.ANNOTATION_TYPE)
@Retention(RetentionPolicy.RUNTIME)
public @interface License {

/**
 * The name of the license.
 */
 String name();

/**
 * An optional URL for the license.
 */
 String url() default "";
}

1.118 accessors-smart 1.2
1.118.1 Available under license :
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2011 JSON-SMART authors
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
*/
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1177937262_1625568439.63/0/accessors-smart-1-2-sources-5-
  jar/net/minidev/asm/BeansAccess.java
* /opt/cola/permits/1177937262_1625568439.63/0/accessors-smart-1-2-sources-5-
  jar/net/minidev/asm/DynamicClassLoader.java
* /opt/cola/permits/1177937262_1625568439.63/0/accessors-smart-1-2-sources-5-
  jar/net/minidev/asm/Accessor.java
* /opt/cola/permits/1177937262_1625568439.63/0/accessors-smart-1-2-sources-5-
  jar/net/minidev/asm/ASMUtil.java

1.119 dnsjava 2.1.7

1.119.1 Available under license:

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the
following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following
disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following
disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR
ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS
OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN
IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.120 jacoco-core 0.8.5
1.120.1 Available under license:
Found license 'Eclipse Public License 1.0' in 'under the terms and conditions of the Eclipse Public License Version 2.0'
Found license 'Eclipse Public License 1.0' in '* This program and the accompanying materials are made available under * the terms of the Eclipse Public License 2.0 which is available at'

1.121 commons-configuration 1.10
1.121.1 Available under license:
Apache Commons Configuration
Copyright 2001-2013 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.
5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.122 asm-analysis 7.1
1.122.1 Available under license:
No license file was found, but licenses were detected in source scan.

2011 INRIA, France Telecom
* All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
*    notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
*    notice, this list of conditions and the following disclaimer in the
*    documentation and/or other materials provided with the distribution.
* 3. Neither the name of the copyright holders nor the names of its
*    contributors may be used to endorse or promote products derived from
*    this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
* LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
* CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
* SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
* INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
* CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
* THE POSSIBILITY OF SUCH DAMAGE.

Found in path(s):
* /opt/cola/permits/1135892660_1613626660.05/0/asm-analysis-7-1-sources-2-
  jar/org/objectweb/asm/tree/analysis/package.html
No license file was found, but licenses were detected in source scan.

// All rights reserved.
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions
// are met:
// 1. Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
// 2. Redistributions in binary form must reproduce the above copyright
// notice, this list of conditions and the following disclaimer in the
// documentation and/or other materials provided with the distribution.
// 3. Neither the name of the copyright holders nor the names of its
// this software without specific prior written permission.
/**
 * Constructs a new {@link SourceInterpreter} for the latest ASM API version. <i>Subclasses must
 * not use this constructor</i>. Instead, they must use the {@link SourceInterpreter(int)}
 * version.
 */

Found in path(s):
* /opt/cola/permits/1135892660_1613626660.05/0/asm-analysis-7-1-sources-2-
  jar/org/objectweb/asm/tree/analysis/SourceInterpreter.java
No license file was found, but licenses were detected in source scan.

// All rights reserved.
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions
// are met:
// 1. Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
// 2. Redistributions in binary form must reproduce the above copyright
// notice, this list of conditions and the following disclaimer in the
// documentation and/or other materials provided with the distribution.
// 3. Neither the name of the copyright holders nor the names of its
// this software without specific prior written permission.

Found in path(s):
* /opt/cola/permits/1135892660_1613626660.05/0/asm-analysis-7-1-sources-2-jar/org/objectweb/asm/tree/analysis/SourceValue.java
* /opt/cola/permits/1135892660_1613626660.05/0/asm-analysis-7-1-sources-2-jar/org/objectweb/asm/tree/analysis/SmallSet.java
* /opt/cola/permits/1135892660_1613626660.05/0/asm-analysis-7-1-sources-2-jar/org/objectweb/asm/tree/analysis/BasicVerifier.java
* /opt/cola/permits/1135892660_1613626660.05/0/asm-analysis-7-1-sources-2-jar/org/objectweb/asm/tree/analysis/Subroutine.java
* /opt/cola/permits/1135892660_1613626660.05/0/asm-analysis-7-1-sources-2-jar/org/objectweb/asm/tree/analysis/BasicValue.java
* /opt/cola/permits/1135892660_1613626660.05/0/asm-analysis-7-1-sources-2-jar/org/objectweb/asm/tree/analysis/AnalyzerException.java
* /opt/cola/permits/1135892660_1613626660.05/0/asm-analysis-7-1-sources-2-jar/org/objectweb/asm/tree/analysis/Frame.java
* /opt/cola/permits/1135892660_1613626660.05/0/asm-analysis-7-1-sources-2-jar/org/objectweb/asm/tree/analysis/SimpleVerifier.java
* /opt/cola/permits/1135892660_1613626660.05/0/asm-analysis-7-1-sources-2-jar/org/objectweb/asm/tree/analysis/Value.java
* /opt/cola/permits/1135892660_1613626660.05/0/asm-analysis-7-1-sources-2-jar/org/objectweb/asm/tree/analysis/Interpreter.java
* /opt/cola/permits/1135892660_1613626660.05/0/asm-analysis-7-1-sources-2-jar/org/objectweb/asm/tree/analysis/Analyzer.java

No license file was found, but licenses were detected in source scan.

// All rights reserved.
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions
// are met:
// 1. Redistributions of source code must retain the above copyright
//    notice, this list of conditions and the following disclaimer.
// 2. Redistributions in binary form must reproduce the above copyright
//    notice, this list of conditions and the following disclaimer in the
//    documentation and/or other materials provided with the distribution.
// 3. Neither the name of the copyright holders nor the names of its
//    this software without specific prior written permission.
/**
 * Constructs a new {@link BasicInterpreter} for the latest ASM API version. <i>Subclasses must
 * not use this constructor</i>. Instead, they must use the {@link BasicInterpreter(int)}
 * version.
 */

Found in path(s):
* /opt/cola/permits/1135892660_1613626660.05/0/asm-analysis-7-1-sources-2-jar/org/objectweb/asm/tree/analysis/BasicInterpreter.java
1.123 jedis 3.3.0
1.123.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2009-2010 MBTE Sweden AB. Licensed under the Apache License, Version 2.0 (the
 * "License"); you may not use this file except in compliance with the License. You may obtain a
 * copy of the License at http://www.apache.org/licenses/LICENSE-2.0 Unless required by applicable
 * law or agreed to in writing, software distributed under the License is distributed on an "AS IS"
 * BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License
 * for the specific language governing permissions and limitations under the License.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1047560748_1590497249.67/0/jedis-3-3-0-sources-
  jar/redis/clients/jedis/util/RedisInputStream.java
No license file was found, but licenses were detected in source scan.

/*
 * Licensed to the Apache Software Foundation (ASF) under one or more contributor license
 * agreements. See the NOTICE file distributed with this work for additional information regarding
 * copyright ownership. The ASF licenses this file to You under the Apache License, Version 2.0 (the
 * "License"); you may not use this file except in compliance with the License. You may obtain a
 * copy of the License at http://www.apache.org/licenses/LICENSE-2.0 Unless required by applicable
 * law or agreed to in writing, software distributed under the License is distributed on an "AS IS"
 * BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License
 * for the specific language governing permissions and limitations under the License.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1047560748_1590497249.67/0/jedis-3-3-0-sources-
  jar/redis/clients/jedis/util/MurmurHash.java

1.124 jackson-mapper-asl 1.9.2
1.124.1 Available under license:

No license file was found, but licenses were detected in source scan.

/* Jackson JSON-processor.
 *
 * Copyright (c) 2007- Tatu Saloranta, tatu.saloranta@iki.fi
 */
* Licensed under the License specified in file LICENSE, included with
* the source code and binary code bundles.
* You may not use this file except in compliance with the License.
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
* /

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1078377000_1597297561.19/0/jackson-mapper-asl-1-9-2-sources-1-
  jar/org/codehaus/jackson/map/MappingJsonFactory.java

1.125 log4j-api 2.17.0

1.125.1 Available under license:

Apache Log4j 1.x Compatibility API
Copyright 1999-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.
"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.126 httpcomponents-client 4.5.13

1.126.1 Available under license:

Apache HttpComponents Client
Copyright 1999-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

=========================================================================
This project includes Public Suffix List copied from
<https://publicsuffix.org/list/effective_tld_names.dat>
licensed under the terms of the Mozilla Public License, v. 2.0

Full license text: <http://mozilla.org/MPL/2.0/>

Mozilla Public License Version 2.0
==================================

1. Definitions
-------------

1.1. "Contributor" means each individual or legal entity that creates, contributes to the creation of, or owns Covered Software.
1.2. "Contributor Version"
means the combination of the Contributions of others (if any) used by a Contributor and that particular Contributor's Contribution.

1.3. "Contribution"
means Covered Software of a particular Contributor.

1.4. "Covered Software"
means Source Code Form to which the initial Contributor has attached the notice in Exhibit A, the Executable Form of such Source Code Form, and Modifications of such Source Code Form, in each case including portions thereof.

1.5. "Incompatible With Secondary Licenses"
means
(a) that the initial Contributor has attached the notice described in Exhibit B to the Covered Software; or

(b) that the Covered Software was made available under the terms of version 1.1 or earlier of the License, but not also under the terms of a Secondary License.

1.6. "Executable Form"
means any form of the work other than Source Code Form.

1.7. "Larger Work"
means a work that combines Covered Software with other material, in a separate file or files, that is not Covered Software.

1.8. "License"
means this document.

1.9. "Licensable"
means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently, any and all of the rights conveyed by this License.

1.10. "Modifications"
means any of the following:

(a) any file in Source Code Form that results from an addition to, deletion from, or modification of the contents of Covered Software; or

(b) any new file in Source Code Form that contains any Covered Software.
1.11. "Patent Claims" of a Contributor
means any patent claim(s), including without limitation, method,
process, and apparatus claims, in any patent Licensable by such
Contributor that would be infringed, but for the grant of the
License, by the making, using, selling, offering for sale, having
made, import, or transfer of either its Contributions or its
Contributor Version.

1.12. "Secondary License"
means either the GNU General Public License, Version 2.0, the GNU
Lesser General Public License, Version 2.1, the GNU Affero General
Public License, Version 3.0, or any later versions of those
licenses.

1.13. "Source Code Form"
means the form of the work preferred for making modifications.

1.14. "You" (or "Your")
means an individual or a legal entity exercising rights under this
License. For legal entities, "You" includes any entity that
controls, is controlled by, or is under common control with You. For
purposes of this definition, "control" means (a) the power, direct
or indirect, to cause the direction or management of such entity,
whether by contract or otherwise, or (b) ownership of more than
fifty percent (50%) of the outstanding shares or beneficial
ownership of such entity.

2. License Grants and Conditions
--------------------------------

2.1. Grants

Each Contributor hereby grants You a world-wide, royalty-free,
non-exclusive license:

(a) under intellectual property rights (other than patent or trademark)
Licensable by such Contributor to use, reproduce, make available,
modify, display, perform, distribute, and otherwise exploit its
Contributions, either on an unmodified basis, with Modifications, or
as part of a Larger Work; and

(b) under Patent Claims of such Contributor to make, use, sell, offer
for sale, have made, import, and otherwise transfer either its
Contributions or its Contributor Version.

2.2. Effective Date

The licenses granted in Section 2.1 with respect to any Contribution
become effective for each Contribution on the date the Contributor first distributes such Contribution.

2.3. Limitations on Grant Scope

The licenses granted in this Section 2 are the only rights granted under this License. No additional rights or licenses will be implied from the distribution or licensing of Covered Software under this License. Notwithstanding Section 2.1(b) above, no patent license is granted by a Contributor:

(a) for any code that a Contributor has removed from Covered Software;

or

(b) for infringements caused by: (i) Your and any other third party's modifications of Covered Software, or (ii) the combination of its Contributions with other software (except as part of its Contributor Version); or

(c) under Patent Claims infringed by Covered Software in the absence of its Contributions.

This License does not grant any rights in the trademarks, service marks, or logos of any Contributor (except as may be necessary to comply with the notice requirements in Section 3.4).

2.4. Subsequent Licenses

No Contributor makes additional grants as a result of Your choice to distribute the Covered Software under a subsequent version of this License (see Section 10.2) or under the terms of a Secondary License (if permitted under the terms of Section 3.3).

2.5. Representation

Each Contributor represents that the Contributor believes its Contributions are its original creation(s) or it has sufficient rights to grant the rights to its Contributions conveyed by this License.

2.6. Fair Use

This License is not intended to limit any rights You have under applicable copyright doctrines of fair use, fair dealing, or other equivalents.

2.7. Conditions

Sections 3.1, 3.2, 3.3, and 3.4 are conditions of the licenses granted
3. Responsibilities

-------------------

3.1. Distribution of Source Form

All distribution of Covered Software in Source Code Form, including any Modifications that You create or to which You contribute, must be under the terms of this License. You must inform recipients that the Source Code Form of the Covered Software is governed by the terms of this License, and how they can obtain a copy of this License. You may not attempt to alter or restrict the recipients' rights in the Source Code Form.

3.2. Distribution of Executable Form

If You distribute Covered Software in Executable Form then:

(a) such Covered Software must also be made available in Source Code Form, as described in Section 3.1, and You must inform recipients of the Executable Form how they can obtain a copy of such Source Code Form by reasonable means in a timely manner, at a charge no more than the cost of distribution to the recipient; and

(b) You may distribute such Executable Form under the terms of this License, or sublicense it under different terms, provided that the license for the Executable Form does not attempt to limit or alter the recipients' rights in the Source Code Form under this License.

3.3. Distribution of a Larger Work

You may create and distribute a Larger Work under terms of Your choice, provided that You also comply with the requirements of this License for the Covered Software. If the Larger Work is a combination of Covered Software with a work governed by one or more Secondary Licenses, and the Covered Software is not Incompatible With Secondary Licenses, this License permits You to additionally distribute such Covered Software under the terms of such Secondary License(s), so that the recipient of the Larger Work may, at their option, further distribute the Covered Software under the terms of either this License or such Secondary License(s).

3.4. Notices

You may not remove or alter the substance of any license notices (including copyright notices, patent notices, disclaimers of warranty, or limitations of liability) contained within the Source Code Form of
the Covered Software, except that You may alter any license notices to
the extent required to remedy known factual inaccuracies.

3.5. Application of Additional Terms

You may choose to offer, and to charge a fee for, warranty, support,
indemnity or liability obligations to one or more recipients of Covered
Software. However, You may do so only on Your own behalf, and not on
behalf of any Contributor. You must make it absolutely clear that any
such warranty, support, indemnity, or liability obligation is offered by
You alone, and You hereby agree to indemnify every Contributor for any
liability incurred by such Contributor as a result of warranty, support,
indemnity or liability terms You offer. You may include additional
disclaimers of warranty and limitations of liability specific to any
jurisdiction.

4. Inability to Comply Due to Statute or Regulation

If it is impossible for You to comply with any of the terms of this
License with respect to some or all of the Covered Software due to
statute, judicial order, or regulation then You must: (a) comply with
the terms of this License to the maximum extent possible; and (b)
describe the limitations and the code they affect. Such description must
be placed in a text file included with all distributions of the Covered
Software under this License. Except to the extent prohibited by statute
or regulation, such description must be sufficiently detailed for a
recipient of ordinary skill to be able to understand it.

5. Termination

5.1. The rights granted under this License will terminate automatically
if You fail to comply with any of its terms. However, if You become
compliant, then the rights granted under this License from a particular
Contributor are reinstated (a) provisionally, unless and until such
Contributor explicitly and finally terminates Your grants, and (b) on an
ongoing basis, if such Contributor fails to notify You of the
non-compliance by some reasonable means prior to 60 days after You have
come back into compliance. Moreover, Your grants from a particular
Contributor are reinstated on an ongoing basis if such Contributor
notifies You of the non-compliance by some reasonable means, this is the
first time You have received notice of non-compliance with this License
from such Contributor, and You become compliant prior to 30 days after
Your receipt of the notice.

5.2. If You initiate litigation against any entity by asserting a patent
infringement claim (excluding declaratory judgment actions,
counter-claims, and cross-claims) alleging that a Contributor Version
directly or indirectly infringes any patent, then the rights granted to
You by any and all Contributors for the Covered Software under Section
2.1 of this License shall terminate.

5.3. In the event of termination under Sections 5.1 or 5.2 above, all
end user license agreements (excluding distributors and resellers) which
have been validly granted by You or Your distributors under this License
prior to termination shall survive termination.

************************************************************************
*                                                                      *
*  6. Disclaimer of Warranty                                           *
*  -------------------------                                           *
*                                                                      *
*  Covered Software is provided under this License on an "as is"       *
*  basis, without warranty of any kind, either expressed, implied, or *
*  statutory, including, without limitation, warranties that the       *
*  Covered Software is free of defects, merchantable, fit for a         *
*  particular purpose or non-infringing. The entire risk as to the     *
*  quality and performance of the Covered Software is with You.        *
*  (not any Contributor) assume the cost of any necessary servicing,  *
*  repair, or correction. This disclaimer of warranty constitutes an    *
*  essential part of this License. No use of any Covered Software is   *
*  authorized under this License except under this disclaimer.         *
*                                                                      *
************************************************************************
************************************************************************

************************************************************************
*                                                                      *
*  7. Limitation of Liability                                          *
*  --------------------------                                          *
*                                                                      *
*  Under no circumstances and under no legal theory, whether tort      *
*  (including negligence), contract, or otherwise, shall any          *
*  Contributor, or anyone who distributes Covered Software as          *
*  permitted above, be liable to You for any direct, indirect,         *
*  special, incidental, or consequential damages of any character      *
*  including, without limitation, damages for lost profits, loss of   *
*  goodwill, work stoppage, computer failure or malfunction, or any   *
*  and all other commercial damages or losses, even if such party      *
*  shall have been informed of the possibility of such damages. This   *
*  limitation of liability shall not apply to liability for death or   *
*  personal injury resulting from such party’s negligence to the       *
*  extent applicable law prohibits such limitation. Some              *
*  jurisdictions do not allow the exclusion or limitation of           *
*  incidental or consequential damages, so this exclusion and          *
8. Litigation

Any litigation relating to this License may be brought only in the courts of a jurisdiction where the defendant maintains its principal place of business and such litigation shall be governed by laws of that jurisdiction, without reference to its conflict-of-law provisions. Nothing in this Section shall prevent a party's ability to bring cross-claims or counter-claims.

9. Miscellaneous

This License represents the complete agreement concerning the subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not be used to construe this License against a Contributor.

10. Versions of the License

10.1. New Versions

Mozilla Foundation is the license steward. Except as provided in Section 10.3, no one other than the license steward has the right to modify or publish new versions of this License. Each version will be given a distinguishing version number.

10.2. Effect of New Versions

You may distribute the Covered Software under the terms of the version of the License under which You originally received the Covered Software, or under the terms of any subsequent version published by the license steward.

10.3. Modified Versions

If you create software not governed by this License, and you want to create a new license for such software, you may create and use a modified version of this License if you rename the license and remove any references to the name of the license steward (except to note that such modified license differs from this License).
10.4. Distributing Source Code Form that is Incompatible With Secondary Licenses

If You choose to distribute Source Code Form that is Incompatible With Secondary Licenses under the terms of this version of the License, the notice described in Exhibit B of this License must be attached.

Exhibit A - Source Code Form License Notice

This Source Code Form is subject to the terms of the Mozilla Public License, v. 2.0. If a copy of the MPL was not distributed with this file, You can obtain one at http://mozilla.org/MPL/2.0/.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

Exhibit B - "Incompatible With Secondary Licenses" Notice

This Source Code Form is "Incompatible With Secondary Licenses", as defined by the Mozilla Public License, v. 2.0.

1.127 jffi 1.3.0

1.128 uadetector-core 0.9.22

1.128.1 Available under license :

****** Creative Commons Legal Code ******

http://creativecommons.org/licenses/by/3.0/legalcode

****** Attribution 3.0 Unported ******

CREATIVE COMMONS CORPORATION IS NOT A LAW FIRM AND DOES NOT PROVIDE LEGAL SERVICES. DISTRIBUTION OF THIS LICENSE DOES NOT CREATE AN ATTORNEY-CLIENT RELATIONSHIP. CREATIVE COMMONS PROVIDES THIS INFORMATION ON AN "AS-IS" BASIS. CREATIVE COMMONS MAKES NO WARRANTIES REGARDING THE INFORMATION PROVIDED, AND DISCLAIMS LIABILITY FOR DAMAGES RESULTING FROM ITS USE.
**** License ****

THE WORK (AS DEFINED BELOW) IS PROVIDED UNDER THE TERMS OF THIS CREATIVE COMMONS PUBLIC LICENSE ("CCPL" OR "LICENSE"). THE WORK IS PROTECTED BY COPYRIGHT AND/OR OTHER APPLICABLE LAW. ANY USE OF THE WORK OTHER THAN AS AUTHORIZED UNDER THIS LICENSE OR COPYRIGHT LAW IS PROHIBITED. BY EXERCISING ANY RIGHTS TO THE WORK PROVIDED HERE, YOU ACCEPT AND AGREE TO BE BOUND BY THE TERMS OF THIS LICENSE. TO THE EXTENT THIS LICENSE MAY BE CONSIDERED TO BE A CONTRACT, THE LICENSOR GRANTS YOU THE RIGHTS CONTAINED HERE IN CONSIDERATION OF YOUR ACCEPTANCE OF SUCH TERMS AND CONDITIONS.

1. Definitions
   a. "Adaptation" means a work based upon the Work, or upon the Work and other pre-existing works, such as a translation, adaptation, derivative work, arrangement of music or other alterations of a literary or artistic work, or phonogram or performance and includes cinematographic adaptations or any other form in which the Work may be recast, transformed, or adapted including in any form recognizably derived from the original, except that a work that constitutes a Collection will not be considered an Adaptation for the purpose of this License. For the avoidance of doubt, where the Work is a musical work, performance or phonogram, the synchronization of the Work in timed-relation with a moving image ("synching") will be considered an Adaptation for the purpose of this License.
   b. "Collection" means a collection of literary or artistic works, such as encyclopedias and anthologies, or performances, phonograms or broadcasts, or other works or subject matter other than works listed in Section 1(f) below, which, by reason of the selection and arrangement of their contents, constitute intellectual creations, in which the Work is included in its entirety in unmodified form along with one or more other contributions, each constituting separate and independent works in themselves, which together are assembled into a collective whole. A work that constitutes a Collection will not be considered an Adaptation (as defined above) for the purposes of this License.
   c. "Distribute" means to make available to the public the original and copies of the Work or Adaptation, as appropriate, through sale or other transfer of ownership.
   d. "Licensor" means the individual, individuals, entity or entities that offer(s) the Work under the terms of this License.
   e. "Original Author" means, in the case of a literary or artistic work, the individual, individuals, entity or entities who created the Work or if no individual or entity can be identified, the publisher; and in addition (i) in the case of a performance the actors, singers, musicians, dancers, and other persons who act, sing, deliver, declaim, play in, interpret or otherwise perform literary or artistic works or expressions of folklore; (ii) in the case of a phonogram the producer being the person or legal entity who first fixes the sounds of a performance or other sounds; and,
(iii) in the case of broadcasts, the organization that transmits the broadcast.
f. "Work" means the literary and/or artistic work offered under the terms of this License including without limitation any production in the literary, scientific and artistic domain, whatever may be the mode or form of its expression including digital form, such as a book, pamphlet and other writing; a lecture, address, sermon or other work of the same nature; a dramatic or dramatico-musical work; a choreographic work or entertainment in dumb show; a musical composition with or without words; a cinematographic work to which are assimilated works expressed by a process analogous to cinematography; a work of drawing, painting, architecture, sculpture, engraving or lithography; a photographic work to which are assimilated works expressed by a process analogous to photography; a work of applied art; an illustration, map, plan, sketch or three-dimensional work relative to geography, topography, architecture or science; a performance; a broadcast; a phonogram; a compilation of data to the extent it is protected as a copyrightable work; or a work performed by a variety or circus performer to the extent it is not otherwise considered a literary or artistic work.
g. "You" means an individual or entity exercising rights under this License who has not previously violated the terms of this License with respect to the Work, or who has received express permission from the Licensor to exercise rights under this License despite a previous violation.
h. "Publicly Perform" means to perform public recitations of the Work and to communicate to the public those public recitations, by any means or process, including by wire or wireless means or public digital performances; to make available to the public Works in such a way that members of the public may access these Works from a place and at a place individually chosen by them; to perform the Work to the public by any means or process and the communication to the public of the performances of the Work, including by public digital performance; to broadcast and rebroadcast the Work by any means including signs, sounds or images.
i. "Reproduce" means to make copies of the Work by any means including without limitation by sound or visual recordings and the right of fixation and reproducing fixations of the Work, including storage of a protected performance or phonogram in digital form or other electronic medium.

2. Fair Dealing Rights. Nothing in this License is intended to reduce, limit, or restrict any uses free from copyright or rights arising from limitations or exceptions that are provided for in connection with the copyright protection under copyright law or other applicable laws.

3. License Grant. Subject to the terms and conditions of this License, Licensor hereby grants You a worldwide, royalty-free, non-exclusive, perpetual (for the duration of the applicable copyright) license to exercise the rights in the
Work as stated below:

a. to Reproduce the Work, to incorporate the Work into one or more Collections, and to Reproduce the Work as incorporated in the Collections;

b. to create and Reproduce Adaptations provided that any such Adaptation, including any translation in any medium, takes reasonable steps to clearly label, demarcate or otherwise identify that changes were made to the original Work. For example, a translation could be marked "The original work was translated from English to Spanish," or a modification could indicate "The original work has been modified."

c. to Distribute and Publicly Perform the Work including as incorporated in Collections; and,

d. to Distribute and Publicly Perform Adaptations.

e. For the avoidance of doubt:
   i. Non-waivable Compulsory License Schemes. In those jurisdictions in which the right to collect royalties through any statutory or compulsory licensing scheme cannot be waived, the Licensor reserves the exclusive right to collect such royalties for any exercise by You of the rights granted under this License;
   ii. Waivable Compulsory License Schemes. In those jurisdictions in which the right to collect royalties through any statutory or compulsory licensing scheme can be waived, the Licensor waives the exclusive right to collect such royalties for any exercise by You of the rights granted under this License; and,
   iii. Voluntary License Schemes. The Licensor waives the right to collect royalties, whether individually or, in the event that the Licensor is a member of a collecting society that administers voluntary licensing schemes, via that society, from any exercise by You of the rights granted under this License.

The above rights may be exercised in all media and formats whether now known or hereafter devised. The above rights include the right to make such modifications as are technically necessary to exercise the rights in other media and formats. Subject to Section 8(f), all rights not expressly granted by Licensor are hereby reserved.

4. Restrictions. The license granted in Section 3 above is expressly made subject to and limited by the following restrictions:

a. You may Distribute or Publicly Perform the Work only under the terms of this License. You must include a copy of, or the Uniform Resource Identifier (URI) for, this License with every copy of the Work You Distribute or Publicly Perform. You may not offer or impose any terms on the Work that restrict the terms of this License or the ability of the recipient of the Work to exercise the rights granted to that recipient under the terms of the License. You may not sublicense the Work. You must keep intact all notices that refer to this License and to the disclaimer of warranties with every copy of the Work You Distribute or Publicly Perform. When You Distribute or Publicly Perform the Work, You may not
impose any effective technological measures on the Work that restrict the ability of a recipient of the Work from You to exercise the rights granted to that recipient under the terms of the License. This Section 4(a) applies to the Work as incorporated in a Collection, but this does not require the Collection apart from the Work itself to be made subject to the terms of this License. If You create a Collection, upon notice from any Licensor You must, to the extent practicable, remove from the Collection any credit as required by Section 4(b), as requested. If You create an Adaptation, upon notice from any Licensor You must, to the extent practicable, remove from the Adaptation any credit as required by Section 4(b), as requested.

b. If You Distribute, or Publicly Perform the Work or any Adaptations or Collections, You must, unless a request has been made pursuant to Section 4(a), keep intact all copyright notices for the Work and provide, reasonable to the medium or means You are utilizing: (i) the name of the Original Author (or pseudonym, if applicable) if supplied, and/or if the Original Author and/or Licensor designate another party or parties (e.g., a sponsor institute, publishing entity, journal) for attribution ("Attribution Parties") in Licensor's copyright notice, terms of service or by other reasonable means, the name of such party or parties; (ii) the title of the Work if supplied; (iii) to the extent reasonably practicable, the URI, if any, that Licensor specifies to be associated with the Work, unless such URI does not refer to the copyright notice or licensing information for the Work; and (iv), consistent with Section 3(b), in the case of an Adaptation, a credit identifying the use of the Work in the Adaptation (e.g., "French translation of the Work by Original Author," or "Screenplay based on original Work by Original Author"). The credit required by this Section 4(b) may be implemented in any reasonable manner; provided, however, that in the case of a Adaptation or Collection, at a minimum such credit will appear, if a credit for all contributing authors of the Adaptation or Collection appears, then as part of these credits and in a manner at least as prominent as the credits for the other contributing authors. For the avoidance of doubt, You may only use the credit required by this Section for the purpose of attribution in the manner set out above and, by exercising Your rights under this License, You may not implicitly or explicitly assert or imply any connection with, sponsorship or endorsement by the Original Author, Licensor and/or Attribution Parties, as appropriate, of You or Your use of the Work, without the separate, express prior written permission of the Original Author, Licensor and/or Attribution Parties.

c. Except as otherwise agreed in writing by the Licensor or as may be otherwise permitted by applicable law, if You Reproduce, Distribute or Publicly Perform the Work either by itself or as part of any Adaptations or Collections, You must not distort, mutilate, modify or take other derogatory action in relation to the Work which would be prejudicial to the Original Author's honor or reputation. Licensor agrees that in those jurisdictions (e.g. Japan), in which any exercise of the right granted in Section 3(b) of this License (the right to make Adaptations) would be
deemed to be a distortion, mutilation, modification or other derogatory action prejudicial to the Original Author's honor and reputation, the Licensor will waive or not assert, as appropriate, this Section, to the fullest extent permitted by the applicable national law, to enable You to reasonably exercise Your right under Section 3(b) of this License (right to make Adaptations) but not otherwise.

5. Representations, Warranties and Disclaimer

UNLESS OTHERWISE MUTUALLY AGREED TO BY THE PARTIES IN WRITING, LICENSOR OFFERS THE WORK AS-IS AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING THE WORK, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT, OR THE ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY, OR THE PRESENCE OF ABSENCE OF ERRORS, WHETHER OR NOT DISCOVERABLE. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO SUCH EXCLUSION MAY NOT APPLY TO YOU.

EXCEPT TO THE EXTENT REQUIRED BY APPLICABLE LAW, IN NO EVENT WILL LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY FOR ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES ARISING OUT OF THIS LICENSE OR THE USE OF THE WORK, EVEN IF LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. Termination
a. This License and the rights granted hereunder will terminate automatically upon any breach by You of the terms of this License. Individuals or entities who have received Adaptations or Collections from You under this License, however, will not have their licenses terminated provided such individuals or entities remain in full compliance with those licenses. Sections 1, 2, 5, 6, 7, and 8 will survive any termination of this License.

b. Subject to the above terms and conditions, the license granted here is perpetual (for the duration of the applicable copyright in the Work). Notwithstanding the above, Licensor reserves the right to release the Work under different license terms or to stop distributing the Work at any time; provided, however that any such election will not serve to withdraw this License (or any other license that has been, or is required to be, granted under the terms of this License), and this License will continue in full force and effect unless terminated as stated above.

8. Miscellaneous
a. Each time You Distribute or Publicly Perform the Work or a Collection, the Licensor offers to the recipient a license to the Work on the same terms and conditions as the license granted to You under this License.

b. Each time You Distribute or Publicly Perform an Adaptation, Licensor
offers to the recipient a license to the original Work on the same terms and conditions as the license granted to You under this License.
c. If any provision of this License is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this License, and without further action by the parties to this agreement, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.
d. No term or provision of this License shall be deemed waived and no breach consented to unless such waiver or consent shall be in writing and signed by the party to be charged with such waiver or consent.
e. This License constitutes the entire agreement between the parties with respect to the Work licensed here. There are no understandings, agreements or representations with respect to the Work not specified here. Licensor shall not be bound by any additional provisions that may appear in any communication from You. This License may not be modified without the mutual written agreement of the Licensor and You.
f. The rights granted under, and the subject matter referenced, in this License were drafted utilizing the terminology of the Berne Convention for the Protection of Literary and Artistic Works (as amended on September 28, 1979), the Rome Convention of 1961, the WIPO Copyright Treaty of 1996, the WIPO Performances and Phonograms Treaty of 1996 and the Universal Copyright Convention (as revised on July 24, 1971). These rights and subject matter take effect in the relevant jurisdiction in which the License terms are sought to be enforced according to the corresponding provisions of the implementation of those treaty provisions in the applicable national law. If the standard suite of rights granted under applicable copyright law includes additional rights not granted under this License, such additional rights are deemed to be included in the License; this License is not intended to restrict the license of any rights under applicable law.

**** Creative Commons Notice ****

Creative Commons is not a party to this License, and makes no warranty whatsoever in connection with the Work. Creative Commons will not be liable to You or any party on any legal theory for any damages whatsoever, including without limitation any general, special, incidental or consequential damages arising in connection to this license. Notwithstanding the foregoing two (2) sentences, if Creative Commons has expressly identified itself as the Licensor hereunder, it shall have all rights and obligations of Licensor. Except for the limited purpose of indicating to the public that the Work is licensed under the CCPL, Creative Commons does not authorize the use by either party of the trademark "Creative Commons" or any related trademark or logo of Creative Commons without the prior
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License. You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the
Copyright 2012 Andr Roul

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

GNU LESSER GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates
the terms and conditions of version 3 of the GNU General Public
License, supplemented by the additional permissions listed below.

0. Additional Definitions.

As used herein, "this License" refers to version 3 of the GNU Lesser
General Public License, and the "GNU GPL" refers to version 3 of the GNU
General Public License.

"The Library" refers to a covered work governed by this License,
other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided
by the Library, but which is not otherwise based on the Library.
Defining a subclass of a class defined by the Library is deemed a mode
of using an interface provided by the Library.

A "Combined Work" is a work produced by combining or linking an
Application with the Library. The particular version of the Library
with which the Combined Work was made is also called the "Linked
Version".

The "Minimal Corresponding Source" for a Combined Work means the
Corresponding Source for the Combined Work, excluding any source code for portions of the Combined Work that, considered in isolation, are based on the Application, and not on the Linked Version.

The "Corresponding Application Code" for a Combined Work means the object code and/or source code for the Application, including any data and utility programs needed for reproducing the Combined Work from the Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or

b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.


The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the object code with a copy of the GNU GPL and this license document.

You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of the following:

a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the Combined Work with a copy of the GNU GPL and this license document.

c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document.

d) Do one of the following:

0) Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to recombine or relink the Application with a modified version of the Linked Version to produce a modified Combined Work, in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.

1) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (a) uses at run time a copy of the Library already present on the user's computer system, and (b) will operate properly with a modified version of the Library that is interface-compatible with the Linked Version.

e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)

You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

The UADetector Project
-----------------------

Please visit the UADetector web site for more information:

* http://uadetector.sourceforge.net/

Copyright 2012 Andr Roul

Andr Roul licenses this product to you under the Apache License, version 2.0
(the "License"); you may not use this product except in compliance with the License. You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Also, please refer to each LICENSE.<component>.txt file, which is located in the same directory as this file, for the license terms of the components that this product depends on.

------------------------------------------------------------------------------------------------------------------------
This product contains a modified version of Dave Koelle's Alphanum Algorithm, which can be obtained at:

* LICENSE:
  * LICENSE.alphanum.txt (GNU LGPL 2.1 or above)
* HOMEPAGE:
  * http://www.davekoelle.com/alphanum.html

This product uses a version of Jaroslav Mallat's UAS Data, which can be obtained at:

* LICENSE:
  * LICENSE.uas.txt (CC BY 3.0)
* HOMEPAGE:
  * http://user-agent-string.info/

1.129 spring-cloud-starter-consul-config 3.0.4
1.129.1 Available under license:

No license file was found, but licenses were detected in source scan.

<name>Apache License, Version 2.0</name>
Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at distributed under the License is distributed on an "AS IS" BASIS.

Found in path(s):
* /opt/cola/permits/1222421669_1644998496.95/0/spring-cloud-starter-consul-config-3.0.4-jar/META-INF/maven/org.springframework.cloud/spring-cloud-starter-consul-config/pom.xml
1.130 jersey-server 1.19.1

1.131 classgraph 4.8.21

1.131.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * This file is part of ClassGraph.
 * 
 * Author: Luke Hutchison (luke.hutch@gmail.com)
 * 
 * Hosted at: https://github.com/classgraph/classgraph
 * 
 * The MIT License (MIT)
 * 
 * Copyright (c) 2019 Luke Hutchison
 * 
 * Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
 * documentation files (the "Software"), to deal in the Software without restriction, including without
 * limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of
 * the Software, and to permit persons to whom the Software is furnished to do so, subject to the following
 * conditions:
 * 
 * The above copyright notice and this permission notice shall be included in all copies or substantial
 * portions of the Software.
 * 
 * THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
 * IMPLIED, INCLUDING BUT NOT
 * LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE
 * AND NONINFRINGEMENT. IN NO
 * EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES
 * OR OTHER LIABILITY, WHETHER IN
 * AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
 * WITH THE SOFTWARE OR THE USE
 * OR OTHER DEALINGS IN THE SOFTWARE.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1023248131_1591358362.39/0/classgraph-4-8-21-sources-jar/io/github/classgraph/ResourceList.java

No license file was found, but licenses were detected in source scan.
/*
 * This file is part of ClassGraph.
 *
 * Author: Luke Hutchison
 *
 * Hosted at: https://github.com/classgraph/classgraph
 *
 * --
 *
 * The MIT License (MIT)
 *
 * Copyright (c) 2019 Luke Hutchison
 *
 * Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
 * documentation files (the "Software"), to deal in the Software without restriction, including without
 * limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of
 * the Software, and to permit persons to whom the Software is furnished to do so, subject to the following
 * conditions:
 *
 * The above copyright notice and this permission notice shall be included in all copies or substantial
 * portions of the Software.
 *
 * THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT
 * LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO
 * EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
 * AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE
 * OR OTHER DEALINGS IN THE SOFTWARE.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1023248131_1591358362.39/0/classgraph-4-8-21-sources-jar/io/github/classgraph/MethodParameterInfo.java
* /opt/ws_local/PERMITS_SQL/1023248131_1591358362.39/0/classgraph-4-8-21-sources-jar/io/github/classgraph/ScanResult.java
* /opt/ws_local/PERMITS_SQL/1023248131_1591358362.39/0/classgraph-4-8-21-sources-jar/io/github/classgraph/ScanResultObject.java
* /opt/ws_local/PERMITS_SQL/1023248131_1591358362.39/0/classgraph-4-8-21-sources-jar/nonapi/io/github/classgraph/fastzipfilereader/RecyclableInflater.java
* /opt/ws_local/PERMITS_SQL/1023248131_1591358362.39/0/classgraph-4-8-21-sources-jar/nonapi/io/github/classgraph/utils/InputStreamOrByteBufferAdapter.java
* /opt/ws_local/PERMITS_SQL/1023248131_1591358362.39/0/classgraph-4-8-21-sources-jar/io/github/classgraph/ModuleInfo.java
* /opt/ws_local/PERMITS_SQL/1023248131_1591358362.39/0/classgraph-4-8-21-sources-jar/io/github/classgraph/BaseTypeSignature.java
Open Source Used In security_automation identity-guest-cs 699

*/opt/ws_local/PERMITS_SQL/1023248131_1591358362.39/0/classgraph-4-8-21-sources-jar/nonapi/io/github/classgraph/classloaderhandler/EquinoxContextFinderClassLoaderHandler.java
*/opt/ws_local/PERMITS_SQL/1023248131_1591358362.39/0/classgraph-4-8-21-sources-jar/io/github/classgraph/ObjectTypedValueWrapper.java
*/opt/ws_local/PERMITS_SQL/1023248131_1591358362.39/0/classgraph-4-8-21-sources-jar/io/github/classgraph/ClasspathElementTypeModule.java
*/opt/ws_local/PERMITS_SQL/1023248131_1591358362.39/0/classgraph-4-8-21-sources-jar/io/github/classgraph/AnnotationParameterValueList.java
*/opt/ws_local/PERMITS_SQL/1023248131_1591358362.39/0/classgraph-4-8-21-sources-jar/nonapi/io/github/classgraph/object/ReferenceEqualityKey.java
*/opt/ws_local/PERMITS_SQL/1023248131_1591358362.39/0/classgraph-4-8-21-sources-jar/nonapi/io/github/classgraph/types/ParseException.java
*/opt/ws_local/PERMITS_SQL/1023248131_1591358362.39/0/classgraph-4-8-21-sources-jar/io/github/classgraph/TypeArgument.java
*/opt/ws_local/PERMITS_SQL/1023248131_1591358362.39/0/classgraph-4-8-21-sources-jar/nonapi/io/github/classgraph/ClassFields.java
*/opt/ws_local/PERMITS_SQL/1023248131_1591358362.39/0/classgraph-4-8-21-sources-jar/nonapi/io/github/classgraph/concurrency/InterruptionChecker.java
*/opt/ws_local/PERMITS_SQL/1023248131_1591358362.39/0/classgraph-4-8-21-sources-jar/io/github/classgraph/MethodInfo.java
*/opt/ws_local/PERMITS_SQL/1023248131_1591358362.39/0/classgraph-4-8-21-sources-jar/io/github/classgraph/AnnotationClassRef.java
*/opt/ws_local/PERMITS_SQL/1023248131_1591358362.39/0/classgraph-4-8-21-sources-jar/io/github/classgraph/ClassRefTypeSignature.java
*/opt/ws_local/PERMITS_SQL/1023248131_1591358362.39/0/classgraph-4-8-21-sources-jar/nonapi/io/github/classgraph/utils/URLPathEncoder.java
*/opt/ws_local/PERMITS_SQL/1023248131_1591358362.39/0/classgraph-4-8-21-sources-jar/nonapi/io/github/classgraph/ScanSpec.java
*/opt/ws_local/PERMITS_SQL/1023248131_1591358362.39/0/classgraph-4-8-21-sources-jar/nonapi/io/github/classgraph/concurrency/SingletonMap.java
*/opt/ws_local/PERMITS_SQL/1023248131_1591358362.39/0/classgraph-4-8-21-sources-jar/io/github/classgraph/MappableColumnInfoList.java
*/opt/ws_local/PERMITS_SQL/1023248131_1591358362.39/0/classgraph-4-8-21-sources-jar/io/github/classgraph/HasName.java
*/opt/ws_local/PERMITS_SQL/1023248131_1591358362.39/0/classgraph-4-8-21-sources-jar/io/github/classgraph/ModuleRef.java
*/opt/ws_local/PERMITS_SQL/1023248131_1591358362.39/0/classgraph-4-8-21-sources-jar/nonapi/io/github/classgraph/json/ClassFieldCache.java
*/opt/ws_local/PERMITS_SQL/1023248131_1591358362.39/0/classgraph-4-8-21-sources-jar/nonapi/io/github/classgraph/classloaderhandler/EquinoxClassLoaderHandler.java
*/opt/ws_local/PERMITS_SQL/1023248131_1591358362.39/0/classgraph-4-8-21-sources-jar/io/github/classgraph/ClassGraphException.java
*/opt/ws_local/PERMITS_SQL/1023248131_1591358362.39/0/classgraph-4-8-21-sources-jar/nonapi/io/github/classgraph/fastzipfilereader/NestedJarHandler.java
*/opt/ws_local/PERMITS_SQL/1023248131_1591358362.39/0/classgraph-4-8-21-sources-jar/io/github/classgraph/ClassTypeSignature.java
*/opt/ws_local/PERMITS_SQL/1023248131_1591358362.39/0/classgraph-4-8-21-sources-jar/nonapi/io/github/classgraph/classpath/ClassLoaderAndModuleFinder.java
* /opt/ws_local/PERMITS_SQL/1023248131_1591358362.39/0/classgraph-4-8-21-sources-jar/nonapi/io/github/classgraph/classpath/ClassLoaderOrder.java
* /opt/ws_local/PERMITS_SQL/1023248131_1591358362.39/0/classgraph-4-8-21-sources-jar/nonapi/io/github/classgraph/classloaderhandler/ClassLoaderHandler.java
* /opt/ws_local/PERMITS_SQL/1023248131_1591358362.39/0/classgraph-4-8-21-sources-jar/io/github/classgraph/ModulePathInfo.java
* /opt/ws_local/PERMITS_SQL/1023248131_1591358362.39/0/classgraph-4-8-21-sources-jar/io/github/classgraph/ClasspathElement.java
* /opt/ws_local/PERMITS_SQL/1023248131_1591358362.39/0/classgraph-4-8-21-sources-jar/io/github/classgraph/AnnotationEnumValue.java
* /opt/ws_local/PERMITS_SQL/1023248131_1591358362.39/0/classgraph-4-8-21-sources-jar/io/github/classgraph/ClassGraphClassLoader.java
* /opt/ws_local/PERMITS_SQL/1023248131_1591358362.39/0/classgraph-4-8-21-sources-jar/nonapi/io/github/classgraph/classloaderhandler/URLClassLoaderHandler.java
* /opt/ws_local/PERMITS_SQL/1023248131_1591358362.39/0/classgraph-4-8-21-sources-jar/nonapi/io/github/classgraph/classloaderhandler/PlexusClassWorldsClassRealmClassLoaderHandler.java
* /opt/ws_local/PERMITS_SQL/1023248131_1591358362.39/0/classgraph-4-8-21-sources-jar/nonapi/io/github/classgraph/utils/Join.java
* /opt/ws_local/PERMITS_SQL/1023248131_1591358362.39/0/classgraph-4-8-21-sources-jar/io/github/classgraph/ModuleInfoList.java
* /opt/ws_local/PERMITS_SQL/1023248131_1591358362.39/0/classgraph-4-8-21-sources-jar/io/github/classgraph/AnnotationInfo.java
* /opt/ws_local/PERMITS_SQL/1023248131_1591358362.39/0/classgraph-4-8-21-sources-jar/nonapi/io/github/classgraph/classpath/CallStackReader.java
* /opt/ws_local/PERMITS_SQL/1023248131_1591358362.39/0/classgraph-4-8-21-sources-jar/nonapi/io/github/classgraph/classloaderhandler/IPMSClassLoaderHandler.java
* /opt/ws_local/PERMITS_SQL/1023248131_1591358362.39/0/classgraph-4-8-21-sources-jar/nonapi/io/github/classgraph/utils/ReflectionUtils.java
* /opt/ws_local/PERMITS_SQL/1023248131_1591358362.39/0/classgraph-4-8-21-sources-jar/io/github/classgraph/PackageInfoList.java
* /opt/ws_local/PERMITS_SQL/1023248131_1591358362.39/0/classgraph-4-8-21-sources-jar/io/github/classgraph/json/Id.java
* /opt/ws_local/PERMITS_SQL/1023248131_1591358362.39/0/classgraph-4-8-21-sources-jar/io/github/classgraph/ClassInfo.java
* /opt/ws_local/PERMITS_SQL/1023248131_1591358362.39/0/classgraph-4-8-21-sources-jar/io/github/classgraph/FieldInfo.java
* /opt/ws_local/PERMITS_SQL/1023248131_1591358362.39/0/classgraph-4-8-21-sources-jar/nonapi/io/github/classgraph/json/ParameterizedTypeImpl.java
* /opt/ws_local/PERMITS_SQL/1023248131_1591358362.39/0/classgraph-4-8-21-sources-jar/nonapi/io/github/classgraph/json/JSONArray.java
* /opt/ws_local/PERMITS_SQL/1023248131_1591358362.39/0/classgraph-4-8-21-sources-jar/io/github/classgraph/GraphvizDotfileGenerator.java
* /opt/ws_local/PERMITS_SQL/1023248131_1591358362.39/0/classgraph-4-8-21-sources-jar/nonapi/io/github/classgraph/recycler/Recycler.java
/*
 * This file is part of ClassGraph.
 *
 * Author: Sergey Bespalov
 *
 * Hosted at: https://github.com/classgraph/classgraph
 *
 * --
 *
 * The MIT License (MIT)
 *
 * Copyright (c) 2017 Sergey Bespalov
 *
 * Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
 * documentation files (the "Software"), to deal in the Software without restriction, including without
 * limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of
 * the Software, and to permit persons to whom the Software is furnished to do so, subject to the following
 * conditions:
 *
 * The above copyright notice and this permission notice shall be included in all copies or substantial
 * portions of the Software.
 *
 * THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
 * IMPLIED, INCLUDING BUT NOT
 * LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE
 * AND NONINFRINGEMENT. IN NO
 * EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES
 * OR OTHER LIABILITY, WHETHER IN
 * AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
 * WITH THE SOFTWARE OR THE USE
 * OR OTHER DEALINGS IN THE SOFTWARE.
 */

Found in path(s):
/*
 * This file is part of ClassGraph.
 *
 * Author: @jacobg on GitHub
 *
 * Hosted at: https://github.com/classgraph/classgraph
 *
 * The MIT License (MIT)
 *
 * Copyright (c) 2019 @jacobg, Luke Hutchison
 *
 * Permission is hereby granted, free of charge, to any person obtaining a copy of this software and
 * documentation files (the "Software"), to deal in the Software without restriction, including without
 * limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of
 * the Software, and to permit persons to whom the Software is furnished to do so, subject to the following
 * conditions:
 *
 * The above copyright notice and this permission notice shall be included in all copies or substantial
 * portions of the Software.
 *
 * THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
 * IMPLIED, INCLUDING BUT NOT
 * LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE
 * AND NONINFRINGEMENT. IN NO
 * EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES
 * OR OTHER LIABILITY, WHETHER IN
 * AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
 * WITH THE SOFTWARE OR THE USE
 * OR OTHER DEALINGS IN THE SOFTWARE.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1023248131_1591358362.39/0/classgraph-4-8-21-sources-jar/nonapi/io/github/classgraph/classloaderhandler/AntClassLoaderHandler.java
No license file was found, but licenses were detected in source scan.

<name>The MIT License (MIT)</name>

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1023248131_1591358362.39/0/classgraph-4-8-21-sources-jar/META-INF/maven/io.github.classgraph/classgraph/pom.xml
No license file was found, but licenses were detected in source scan.

/*
 * This file is part of ClassGraph.
 *
 * Author: Luke Hutchison
 *
 * Hosted at: https://github.com/classgraph/classgraph
 *
 * --
 *
 * The MIT License (MIT)
 *
 * Copyright (c) 2019 Luke Hutchison, with significant contributions from Davy De Durpel
 *
 * Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
 * documentation files (the "Software"), to deal in the Software without restriction, including without
 * limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of
 * the Software, and to permit persons to whom the Software is furnished to do so, subject to the following
 * conditions:
 *
 * The above copyright notice and this permission notice shall be included in all copies or substantial
 * portions of the Software.
 *
 * THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
 * IMPLIED, INCLUDING BUT NOT
 * LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE
 * AND NONINFRINGEMENT. IN NO
 * EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES
 * OR OTHER LIABILITY, WHETHER IN
 * AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
 * WITH THE SOFTWARE OR THE USE
 * OR OTHER DEALINGS IN THE SOFTWARE.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1023248131_1591358362.39/0/classgraph-4-8-21-sources-jar/nonapi/io/github/classgraph/classloaderhandler/JBossClassLoaderHandler.java

No license file was found, but licenses were detected in source scan.

/*
 * This file is part of ClassGraph.
 *
 * Author: Michael J. Simons
 *
 * Hosted at: https://github.com/classgraph/classgraph
 *
 * --

Open Source Used In security_automation identity-guest-cs 705
* The MIT License (MIT)
* 
* Copyright (c) 2019 Luke Hutchison
* 
* Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without
limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of
the Software, and to permit persons to whom the Software is furnished to do so, subject to the following
* conditions:
* 
* The above copyright notice and this permission notice shall be included in all copies or substantial
portions of the Software.
* 
* THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT
LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE
AND NONINFRINGEMENT. IN NO
EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES
OR OTHER LIABILITY. WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE
* OR OTHER DEALINGS IN THE SOFTWARE.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1023248131_1591358362.39/0/classgraph-4-8-21-sources-
jar/nonapi/io/github/classgraph/classloaderhandler/SpringBootRestartClassLoaderHandler.java
No license file was found, but licenses were detected in source scan.

/*
* This file is part of ClassGraph.
* 
* Author: Harith Elrufaie
* 
* Hosted at: https://github.com/classgraph/classgraph
* 
* --
* 
* The MIT License (MIT)
* 
* Copyright (c) 2017 Harith Elrufaie
* 
* Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without
limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of
the Software, and to permit persons to whom the Software is furnished to do so, subject to the following
* conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Found in path(s):
/opt/ws_local/PERMITS_SQL/1023248131_1591358362.39/0/classgraph-4-8-21-sources-jar/nonapi/io/github/classgraph/classloaderhandler/FelixClassLoaderHandler.java
No license file was found, but licenses were detected in source scan.

This file is part of ClassGraph.

Author: Johno Crawford (johno@sulake.com)

Hosted at: https://github.com/classgraph/classgraph

The MIT License (MIT)

Copyright (c) 2016 Johno Crawford

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
* AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE
* OR OTHER DEALINGS IN THE SOFTWARE.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1023248131_1591358362.39/0/classgraph-4-8-21-sources-jar/nonapi/io/github/classgraph/concurrency/AutoCloseableExecutorService.java
* /opt/ws_local/PERMITS_SQL/1023248131_1591358362.39/0/classgraph-4-8-21-sources-jar/nonapi/io/github/classgraph/concurrency/SimpleThreadFactory.java

No license file was found, but licenses were detected in source scan.

/*
* This file is part of ClassGraph.
*
* Author: R. Kempees
*
* Hosted at: https://github.com/classgraph/classgraph
*
* The MIT License (MIT)
*
* Copyright (c) 2017 R. Kempees
*
* Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without
limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of
the Software, and to permit persons to whom the Software is furnished to do so, subject to the following
conditions:
*
* The above copyright notice and this permission notice shall be included in all copies or substantial
portions of the Software.
*
* THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT
* LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE
AND NONINFRINGEMENT. IN NO
* EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES
OR OTHER LIABILITY, WHETHER IN
* AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE
* OR OTHER DEALINGS IN THE SOFTWARE.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1023248131_1591358362.39/0/classgraph-4-8-21-sources-jar/nonapi/io/github/classgraph/classloaderhandler/WebsphereLibertyClassLoaderHandler.java
1.132 common-config 5.3.0

1.132.1 Available under license:

No license file was found, but licenses were detected in source scan.

~
~ Copyright 2015 Confluent Inc.
~
~ Licensed under the Apache License, Version 2.0 (the "License");
~ you may not use this file except in compliance with the License.
~ You may obtain a copy of the License at
~
~ http://www.apache.org/licenses/LICENSE-2.0
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS,
~ WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
~ See the License for the specific language governing permissions and
~ limitations under the License.
~

Found in path(s):
* /opt/cola/permits/1257811198_1646348825.22/0/common-config-5-3-0-jar/META-INF/maven/io.confluent/common-config/pom.xml

1.133 j2objc-annotations 1.3

1.134 log4j-over-slf4j 1.7.32

1.134.1 Available under license:

No license file was found, but licenses were detected in source scan.

<url>http://www.apache.org/licenses/LICENSE-2.0.txt</url>

Found in path(s):
* /opt/cola/permits/1198379913_1630347367.42/0/log4j-over-slf4j-1-7-32-jar/META-INF/maven/org.slf4j/log4j-over-slf4j/pom.xml
1.135 netty-transport-native-epoll 4.1.69.Final
1.135.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2016 The Netty Project
 * * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1265673609_1643932648.66/0/netty-transport-native-epoll-4-1-69-final-sources-
jar/io/netty/channel/epoll/LinuxSocket.java
* /opt/cola/permits/1265673609_1643932648.66/0/netty-transport-native-epoll-4-1-69-final-sources-
jar/netty_epoll_linuxsocket.h
* /opt/cola/permits/1265673609_1643932648.66/0/netty-transport-native-epoll-4-1-69-final-sources-
jar/io/netty/channel/epoll/NativeStaticallyReferencedJniMethods.java
* /opt/cola/permits/1265673609_1643932648.66/0/netty-transport-native-epoll-4-1-69-final-sources-
jar/netty_epoll_linuxsocket.c

No license file was found, but licenses were detected in source scan.

~ Copyright 2014 The Netty Project
~
~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~
~ https://www.apache.org/licenses/LICENSE
~ 2.0
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.
Found in path(s):
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2014 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
*  https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/
/**
 * Set the {@code TCP_MD5SIG} option on the socket. See {@code linux/tcp.h} for more details.
 * Keys can only be set on, not read to prevent a potential leak, as they are confidential.
 * Allowing them being read would mean anyone with access to the channel could get them.
 */
/**
 * Set the {@code TCP_QUICKACK} option on the socket.
 * See <a href="https://linux.die.net//man/7/tcp">TCP_QUICKACK</a> for more details.
 */

Found in path(s):
* /opt/cola/permits/1265673609_1643932648.66/0/netty-transport-native-epoll-4-1-69-final-sources.jar/io/netty/channel/epoll/EpollSocketChannelConfig.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2013 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
*  https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1265673609_1643932648.66/0/netty-transport-native-epoll-4-1-69-final-sources-
jar/netty_epoll_native.c
* /opt/cola/permits/1265673609_1643932648.66/0/netty-transport-native-epoll-4-1-69-final-sources-
jar/io/netty/channel/epoll/Native.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1265673609_1643932648.66/0/netty-transport-native-epoll-4-1-69-final-sources-
jar/io/netty/channel/epoll/EpollDomainSocketChannel.java
* /opt/cola/permits/1265673609_1643932648.66/0/netty-transport-native-epoll-4-1-69-final-sources-
jar/io/netty/channel/epoll/EpollServerChannelConfig.java
* /opt/cola/permits/1265673609_1643932648.66/0/netty-transport-native-epoll-4-1-69-final-sources-
jar/io/netty/channel/epoll/TcpMd5Util.java
* /opt/cola/permits/1265673609_1643932648.66/0/netty-transport-native-epoll-4-1-69-final-sources-
jar/io/netty/channel/epoll/EpollChannelConfig.java
* /opt/cola/permits/1265673609_1643932648.66/0/netty-transport-native-epoll-4-1-69-final-sources-
jar/io/netty/channel/epoll/AbstractEpollServerChannel.java
* /opt/cola/permits/1265673609_1643932648.66/0/netty-transport-native-epoll-4-1-69-final-sources-
jar/io/netty/channel/epoll/EpollMode.java
* /opt/cola/permits/1265673609_1643932648.66/0/netty-transport-native-epoll-4-1-69-final-sources-
jar/io/netty/channel/epoll/AbstractEpollStreamChannel.java
* /opt/cola/permits/1265673609_1643932648.66/0/netty-transport-native-epoll-4-1-69-final-sources-
jar/io/netty/channel/epoll/EpollDomainSocketChannelConfig.java
* /opt/cola/permits/1265673609_1643932648.66/0/netty-transport-native-epoll-4-1-69-final-sources-
jar/io/netty/channel/epoll/EpollEventArray.java
* /opt/cola/permits/1265673609_1643932648.66/0/netty-transport-native-epoll-4-1-69-final-sources-
jar/io/netty/channel/epoll/EpollRecvByteAllocatorHandle.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2021 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1265673609_1643932648.66/0/netty-transport-native-epoll-4-1-69-final-sources-jar/io/netty/channel/epoll/EpollDomainDatagramChannel.java
* /opt/cola/permits/1265673609_1643932648.66/0/netty-transport-native-epoll-4-1-69-final-sources-jar/io/netty/channel/epoll/EpollDomainDatagramChannelConfig.java
* /opt/cola/permits/1265673609_1643932648.66/0/netty-transport-native-epoll-4-1-69-final-sources-jar/io/netty/channel/epoll/SegmentedDatagramPacket.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2012 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
/*
 * Copyright 2014 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */
/**
** Set the {@code TCP_MD5SIG} option on the socket. See \{@code linux/tcp.h\} for more details.
* Keys can only be set on, not read to prevent a potential leak, as they are confidential.
* Allowing them being read would mean anyone with access to the channel could get them.
*/

Found in path(s):
* /opt/cola/permits/1265673609_1643932648.66/0/netty-transport-native-epoll-4-1-69-final-sources-
jar/io/netty/channel/epoll/EpollDatagramChannelConfig.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1265673609_1643932648.66/0/netty-transport-native-epoll-4-1-69-final-sources-
jar/io/netty/channel/epoll/EpollServerSocketChannelConfig.java
No license file was found, but licenses were detected in source scan.
1.136 commons-lang3 3.11

1.136.1 Available under license:

Apache Commons Lang
Copyright 2001-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (https://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.
2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or
You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.137 netty-transport-native-unix-common
4.1.69.Final
1.137.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015 The Netty Project
 * 
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * 
 * https://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1265673543_1643922092.65/0/netty-transport-native-unix-common-4.1.69-final-sources-jar/io/netty/channel/unix/FileDescriptor.java
* /opt/cola/permits/1265673543_1643922092.65/0/netty-transport-native-unix-common-4.1.69-final-sources-jar/netty_unix_filedescriptor.h
* /opt/cola/permits/1265673543_1643922092.65/0/netty-transport-native-unix-common-4.1.69-final-sources-jar/io/netty/channel/unix/DatagramSocketAddress.java
* /opt/cola/permits/1265673543_1643922092.65/0/netty-transport-native-unix-common-4.1.69-final-sources-jar/netty_unix_socket.h
* /opt/cola/permits/1265673543_1643922092.65/0/netty-transport-native-unix-common-4.1.69-final-sources-jar/io/netty/channel/unix/ServerDomainSocketChannel.java
* /opt/cola/permits/1265673543_1643922092.65/0/netty-transport-native-unix-common-4.1.69-final-sources-jar/netty_unix_socket.c
* /opt/cola/permits/1265673543_1643922092.65/0/netty-transport-native-unix-common-4.1.69-final-sources-jar/netty_unix_errors.h
* /opt/cola/permits/1265673543_1643922092.65/0/netty-transport-native-unix-common-4.1.69-final-sources-jar/io/netty/channel/unix/NativeInetAddress.java
* /opt/cola/permits/1265673543_1643922092.65/0/netty-transport-native-unix-common-4.1.69-final-sources-jar/io/netty/channel/unix/Errors.java
* /opt/cola/permits/1265673543_1643922092.65/0/netty-transport-native-unix-common-4.1.69-final-sources-jar/netty_unix_errors.c
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2016 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 *   https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2021 The Netty Project
 *
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1265673543_1643922092.65/0/netty-transport-native-unix-common-4-1-69-final-sources-
  jar/io/netty/channel/unix/DomainDatagramSocketAddress.java
* /opt/cola/permits/1265673543_1643922092.65/0/netty-transport-native-unix-common-4-1-69-final-sources-
  jar/io/netty/channel/unix/DomainDatagramPacket.java
* /opt/cola/permits/1265673543_1643922092.65/0/netty-transport-native-unix-common-4-1-69-final-sources-
  jar/io/netty/channel/unix/SegmentedDatagramPacket.java
* /opt/cola/permits/1265673543_1643922092.65/0/netty-transport-native-unix-common-4-1-69-final-sources-
  jar/io/netty/channel/unix/DomainDatagramChannel.java
* /opt/cola/permits/1265673543_1643922092.65/0/netty-transport-native-unix-common-4-1-69-final-sources-
  jar/io/netty/channel/unix/DomainDatagramChannelConfig.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2020 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1265673543_1643922092.65/0/netty-transport-native-unix-common-4-1-69-final-sources-
  jar/netty_unix.h
* /opt/cola/permits/1265673543_1643922092.65/0/netty-transport-native-unix-common-4-1-69-final-sources-
  jar/netty_unix.c
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2018 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1265673543_1643922092.65/0/netty-transport-native-unix-common-4-1-69-final-sources-
  jar/io/netty/channel/unix/Buffer.java
* /opt/cola/permits/1265673543_1643922092.65/0/netty-transport-native-unix-common-4-1-69-final-sources-
  jar/netty_unix_buffer.h
* /opt/cola/permits/1265673543_1643922092.65/0/netty-transport-native-unix-common-4-1-69-final-sources-
  jar/netty_unix_buffer.c
* /opt/cola/permits/1265673543_1643922092.65/0/netty-transport-native-unix-common-4-1-69-final-sources-
  jar/io/netty/channel/unix/PreferredDirectByteBufAllocator.java

No license file was found, but licenses were detected in source scan.

~ Copyright 2016 The Netty Project
~
~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~
~ https://www.apache.org/licenses/LICENSE
2.0
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.

Found in path(s):
* /opt/cola/permits/1265673543_1643922092.65/0/netty-transport-native-unix-common-4-1-69-final-sources-
  jar/META-INF/maven/io.netty/netty-transport-native-unix-common/pom.xml
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2017 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1265673543_1643922092.65/0/netty-transport-native-unix-common-4-1-69-final-sources-
  jar/netty_unix_jni.h
* /opt/cola/permits/1265673543_1643922092.65/0/netty-transport-native-unix-common-4-1-69-final-sources-
  jar/io/netty/channel/unix/UnixChannelUtil.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1265673543_1643922092.65/0/netty-transport-native-unix-common-4-1-69-final-sources-
  jar/io/netty/channel/unix/package-info.java
* /opt/cola/permits/1265673543_1643922092.65/0/netty-transport-native-unix-common-4-1-69-final-sources-
  jar/io/netty/channel/unix/IovArray.java
* /opt/cola/permits/1265673543_1643922092.65/0/netty-transport-native-unix-common-4-1-69-final-sources-
  jar/io/netty/channel/unix/UnixChannelOption.java
1.138 netty-transport-native-kqueue

4.1.69.Final

1.138.1 Available under license:
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2021 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 *   https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1223884368_1635968876.17/0/netty-transport-native-kqueue-4-1-69-final-sources-jar/io/netty/channel/kqueue/KQueueDomainDatagramChannelConfig.java
* /opt/cola/permits/1223884368_1635968876.17/0/netty-transport-native-kqueue-4-1-69-final-sources-jar/io/netty/channel/kqueue/AbstractKQueueDatagramChannel.java
* /opt/cola/permits/1223884368_1635968876.17/0/netty-transport-native-kqueue-4-1-69-final-sources-jar/io/netty/channel/kqueue/KQueueDomainDatagramChannel.java
No license file was found, but licenses were detected in source scan.

/*@
 * Copyright 2016 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 *   https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 */
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1223884368_1635968876.17/0/netty-transport-native-kqueue-4-1-69-final-sources-jar/netty_kqueuebsdsocket.c
* /opt/cola/permits/1223884368_1635968876.17/0/netty-transport-native-kqueue-4-1-69-final-sources-jar/io/netty/channel/kqueue/KQueueServerSocketChannel.java
* /opt/cola/permits/1223884368_1635968876.17/0/netty-transport-native-kqueue-4-1-69-final-sources-jar/io/netty/channel/kqueue/KQueueChannelOption.java
* /opt/cola/permits/1223884368_1635968876.17/0/netty-transport-native-kqueue-4-1-69-final-sources-jar/io/netty/channel/kqueue/AcceptFilter.java
* /opt/cola/permits/1223884368_1635968876.17/0/netty-transport-native-kqueue-4-1-69-final-sources-jar/io/netty/channel/kqueue/KQueueServerDomainSocketChannel.java
* /opt/cola/permits/1223884368_1635968876.17/0/netty-transport-native-kqueue-4-1-69-final-sources-jar/netty_kqueue_eventarray.h
* /opt/cola/permits/1223884368_1635968876.17/0/netty-transport-native-kqueue-4-1-69-final-sources-jar/netty_kqueue_native.c
* /opt/cola/permits/1223884368_1635968876.17/0/netty-transport-native-kqueue-4-1-69-final-sources-jar/io/netty/channel/kqueue/AbstractKQueueChannel.java
* /opt/cola/permits/1223884368_1635968876.17/0/netty-transport-native-kqueue-4-1-69-final-sources-jar/io/netty/channel/kqueue/AbstractKQueueServerChannel.java
* /opt/cola/permits/1223884368_1635968876.17/0/netty-transport-native-kqueue-4-1-69-final-sources-jar/netty_kqueue_eventarray.c
* /opt/cola/permits/1223884368_1635968876.17/0/netty-transport-native-kqueue-4-1-69-final-sources-jar/io/netty/channel/kqueue/Native.java
* /opt/cola/permits/1223884368_1635968876.17/0/netty-transport-native-kqueue-4-1-69-final-sources-jar/io/netty/channel/kqueue/package-info.java
* /opt/cola/permits/1223884368_1635968876.17/0/netty-transport-native-kqueue-4-1-69-final-sources-jar/io/netty/channel/kqueue/KQueueSocketChannel.java
* /opt/cola/permits/1223884368_1635968876.17/0/netty-transport-native-kqueue-4-1-69-final-sources-jar/io/netty/channel/kqueue/KQueueEventArray.java
* /opt/cola/permits/1223884368_1635968876.17/0/netty-transport-native-kqueue-4-1-69-final-sources-jar/io/netty/channel/kqueue/KQueueChannelConfig.java
* /opt/cola/permits/1223884368_1635968876.17/0/netty-transport-native-kqueue-4-1-69-final-sources-jar/io/netty/channel/kqueue/KQueueServerChannelConfig.java
* /opt/cola/permits/1223884368_1635968876.17/0/netty-transport-native-kqueue-4-1-69-final-sources-jar/io/netty/channel/kqueue/KQueueDatagramChannel.java
* /opt/cola/permits/1223884368_1635968876.17/0/netty-transport-native-kqueue-4-1-69-final-sources-jar/io/netty/channel/kqueue/NativeLongArray.java
* /opt/cola/permits/1223884368_1635968876.17/0/netty-transport-native-kqueue-4-1-69-final-sources-jar/io/netty/channel/kqueue/KQueue.java
* /opt/cola/permits/1223884368_1635968876.17/0/netty-transport-native-kqueue-4-1-69-final-sources-jar/io/netty/channel/kqueue/KQueueEventArray.java

No license file was found, but licenses were detected in source scan.

~ Copyright 2016 The Netty Project
~
~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~
~ https://www.apache.org/licenses/LICENSE-2.0

~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.

Found in path(s):

1.139 hystrix-core 1.5.18
1.139.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
   Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
   stating that You changed the files; and
(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0
Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.140 aws-java-sdk-cognitoidp 1.11.868

1.140.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015-2020 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 *
 * Licensed under the Apache License, Version 2.0 (the "License"). You may not use this file except in compliance with
 * the License. A copy of the License is located at
 *
 * http://aws.amazon.com/apache2.0
 *
 * or in the "license" file accompanying this file. This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR
 * CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions
 * and limitations under the License.
 */

Found in path(s):

*/opt/cola/permits/1255512452_1642559226.29/0/aws-java-sdk-cognitoidp-1-11-868-sources-jar/com/amazonaws/services/cognitoidp/model/AccountTakeoverRiskConfigurationType.java
*/opt/cola/permits/1255512452_1642559226.29/0/aws-java-sdk-cognitoidp-1-11-868-sources-jar/com/amazonaws/services/cognitoidp/model/AccountTakeoverRiskConfigurationType.java
*/opt/cola/permits/1255512452_1642559226.29/0/aws-java-sdk-cognitoidp-1-11-868-sources-jar/com/amazonaws/services/cognitoidp/model/transform/CreateUserImportJobResultJsonUnmarshaller.java
*/opt/cola/permits/1255512452_1642559226.29/0/aws-java-sdk-cognitoidp-1-11-868-sources-jar/com/amazonaws/services/cognitoidp/model/transform/CreateUserImportJobResultJsonUnmarshaller.java
*/opt/cola/permits/1255512452_1642559226.29/0/aws-java-sdk-cognitoidp-1-11-868-sources-jar/com/amazonaws/services/cognitoidp/model/CreateUserImportJobResultJsonUnmarshaller.java
/opt/cola/permits/1255512452_1642559226.29/0/aws-java-sdk-cognito-idp-1.11-868-sources.jar/com/amazonaws/services/cognito-idp/model/transform/ContextDataTypeJsonUnmarshaller.java
* /opt/cola/permits/1255512452_1642559226.29/0/aws-java-sdk-cognito-idp-1.11-868-sources.jar/com/amazonaws/services/cognito-idp/model/UnexpectedLambdaException.java
* /opt/cola/permits/1255512452_1642559226.29/0/aws-java-sdk-cognito-idp-1.11-868-sources.jar/com/amazonaws/services/cognito-idp/model/GetUserAttributeVerificationCodeRequest.java
* /opt/cola/permits/1255512452_1642559226.29/0/aws-java-sdk-cognito-idp-1.11-868-sources.jar/com/amazonaws/services/cognito-idp/model/transform/UserPoolPolicyTypeJsonUnmarshaller.java
* /opt/cola/permits/1255512452_1642559226.29/0/aws-java-sdk-cognito-idp-1.11-868-sources.jar/com/amazonaws/services/cognito-idp/model/UpdateUserAttributesRequest.java
* /opt/cola/permits/1255512452_1642559226.29/0/aws-java-sdk-cognito-idp-1.11-868-sources.jar/com/amazonaws/services/cognito-idp/model/transform/DuplicateProviderExceptionUnmarshaller.java
* /opt/cola/permits/1255512452_1642559226.29/0/aws-java-sdk-cognito-idp-1.11-868-sources.jar/com/amazonaws/services/cognito-idp/model/DeleteGroupRequest.java
* /opt/cola/permits/1255512452_1642559226.29/0/aws-java-sdk-cognito-idp-1.11-868-sources.jar/com/amazonaws/services/cognito-idp/model/AttributeDataType.java
* /opt/cola/permits/1255512452_1642559226.29/0/aws-java-sdk-cognito-idp-1.11-868-sources.jar/com/amazonaws/services/cognito-idp/model/transform/AdminDisableUserRequestMarshaller.java
* /opt/cola/permits/1255512452_1642559226.29/0/aws-java-sdk-cognito-idp-1.11-868-sources.jar/com/amazonaws/services/cognito-idp/model/UntagResourceResult.java
* /opt/cola/permits/1255512452_1642559226.29/0/aws-java-sdk-cognito-idp-1.11-868-sources.jar/com/amazonaws/services/cognito-idp/model/SoftwareTokenMfaSettingsType.java
* /opt/cola/permits/1255512452_1642559226.29/0/aws-java-sdk-cognito-idp-1.11-868-sources.jar/com/amazonaws/services/cognito-idp/model/transform/ProviderUserIdentifierTypeJsonUnmarshaller.java
* /opt/cola/permits/1255512452_1642559226.29/0/aws-java-sdk-cognito-idp-1.11-868-sources.jar/com/amazonaws/services/cognito-idp/model/transform/GlobalSignOutResultJsonUnmarshaller.java
* /opt/cola/permits/1255512452_1642559226.29/0/aws-java-sdk-cognito-idp-1.11-868-sources.jar/com/amazonaws/services/cognito-idp/model/transform/InitiateAuthRequestProtocolMarshaller.java
* /opt/cola/permits/1255512452_1642559226.29/0/aws-java-sdk-cognito-idp-1.11-868-sources.jar/com/amazonaws/services/cognito-idp/model/transform/AccountTakeoverActionTypeJsonUnmarshaller.java
* /opt/cola/permits/1255512452_1642559226.29/0/aws-java-sdk-cognito-idp-1.11-868-sources.jar/com/amazonaws/services/cognito-idp/model/transform/SetRiskConfigurationRequestProtocolMarshaller.java
* /opt/cola/permits/1255512452_1642559226.29/0/aws-java-sdk-cognito-idp-1.11-868-sources.jar/com/amazonaws/services/cognito-idp/model/transform/ForgotPasswordRequestProtocolMarshaller.java
* /opt/cola/permits/1255512452_1642559226.29/0/aws-java-sdk-cognito-idp-1.11-868-sources.jar/com/amazonaws/services/cognito-idp/model/transform/ListUserPoolClientsResultJsonUnmarshaller.java
* /opt/cola/permits/1255512452_1642559226.29/0/aws-java-sdk-cognito-idp-1.11-868-sources.jar/com/amazonaws/services/cognito-idp/model/transform/AdminUserGlobalSignOutRequestMarshaller.java
* /opt/cola/permits/1255512452_1642559226.29/0/aws-java-sdk-cognito-idp-1.11-868-sources.jar/com/amazonaws/services/cognito-idp/model/ChallengeResponseType.java

Open Source Used In security_automation identity-guest-cs 734
* /opt/cola/permits/1255512452_1642559226.29/0/aws-java-sdk-cognito-idp-1-11-868-sources-jar/com/amazonaws/services/cognito-idp/model/transform/LimitExceededExceptionUnmarshaller.java
* /opt/cola/permits/1255512452_1642559226.29/0/aws-java-sdk-cognito-idp-1-11-868-sources-jar/com/amazonaws/services/cognito-idp/model/GroupExistsException.java
* /opt/cola/permits/1255512452_1642559226.29/0/aws-java-sdk-cognito-idp-1-11-868-sources-jar/com/amazonaws/services/cognito-idp/model/transform/AttributeTypeJsonUnmarshaller.java
* /opt/cola/permits/1255512452_1642559226.29/0/aws-java-sdk-cognito-idp-1-11-868-sources-jar/com/amazonaws/services/cognito-idp/model/transform/StringAttributeConstraintsTypeMarshaller.java
* /opt/cola/permits/1255512452_1642559226.29/0/aws-java-sdk-cognito-idp-1-11-868-sources-jar/com/amazonaws/services/cognito-idp/model/AdminUpdateDeviceStatusResult.java
jar/com/amazonaws/services/cognitoidp/model/transform/ProviderUserIdentifierTypeMarshaller.java
* /opt/cola/permits/125512452_1642559226.29/0/aws-java-sdk-cognitoidp-1-11-868-sources-jar/com/amazonaws/services/cognitoidp/model/transform/MFAOptionTypeMarshaller.java
* /opt/cola/permits/125512452_1642559226.29/0/aws-java-sdk-cognitoidp-1-11-868-sources-jar/com/amazonaws/services/cognitoidp/model/ResourceNotFoundException.java
* /opt/cola/permits/125512452_1642559226.29/0/aws-java-sdk-cognitoidp-1-11-868-sources-jar/com/amazonaws/services/cognitoidp/model/AdminSetUserMFAPreferenceResult.java
* /opt/cola/permits/125512452_1642559226.29/0/aws-java-sdk-cognitoidp-1-11-868-sources-jar/com/amazonaws/services/cognitoidp/model/transform/AdminForgetDeviceResultJsonUnmarshaller.java
* /opt/cola/permits/125512452_1642559226.29/0/aws-java-sdk-cognitoidp-1-11-868-sources-jar/com/amazonaws/services/cognitoidp/model/DescribeRiskConfigurationResult.java
* /opt/cola/permits/125512452_1642559226.29/0/aws-java-sdk-cognitoidp-1-11-868-sources-jar/com/amazonaws/services/cognitoidp/model/transform/RiskExceptionConfigurationTypeJsonUnmarshaller.java
* /opt/cola/permits/125512452_1642559226.29/0/aws-java-sdk-cognitoidp-1-11-868-sources-jar/com/amazonaws/services/cognitoidp/model/AdminAddUserToGroupResult.java
* /opt/cola/permits/125512452_1642559226.29/0/aws-java-sdk-cognitoidp-1-11-868-sources-jar/com/amazonaws/services/cognitoidp/model/transform/DeleteUserAttributesRequestMarshaller.java
* /opt/cola/permits/125512452_1642559226.29/0/aws-java-sdk-cognitoidp-1-11-868-sources-jar/com/amazonaws/services/cognitoidp/model/InternalErrorException.java
* /opt/cola/permits/125512452_1642559226.29/0/aws-java-sdk-cognitoidp-1-11-868-sources-jar/com/amazonaws/services/cognitoidp/model/AdminInitiateAuthResult.java
* /opt/cola/permits/125512452_1642559226.29/0/aws-java-sdk-cognitoidp-1-11-868-sources-jar/com/amazonaws/services/cognitoidp/model/transform/GlobalSignOutRequestMarshaller.java
* /opt/cola/permits/125512452_1642559226.29/0/aws-java-sdk-cognitoidp-1-11-868-sources-jar/com/amazonaws/services/cognitoidp/model/transform/AccountTakeoverRiskConfigurationTypeJsonUnmarshaller.java
* /opt/cola/permits/125512452_1642559226.29/0/aws-java-sdk-cognitoidp-1-11-868-sources-jar/com/amazonaws/services/cognitoidp/model/transform/ListGroupsRequestProtocolMarshaller.java
* /opt/cola/permits/125512452_1642559226.29/0/aws-java-sdk-cognitoidp-1-11-868-sources-jar/com/amazonaws/services/cognitoidp/model/AdminRespondToAuthChallengeRequest.java
* /opt/cola/permits/125512452_1642559226.29/0/aws-java-sdk-cognitoidp-1-11-868-sources-jar/com/amazonaws/services/cognitoidp/model/AdminUpdateAuthEventFeedbackRequest.java
* /opt/cola/permits/125512452_1642559226.29/0/aws-java-sdk-cognitoidp-1-11-868-sources-jar/com/amazonaws/services/cognitoidp/model/AdminGetUserMFAPreferenceResult.java
* /opt/cola/permits/125512452_1642559226.29/0/aws-java-sdk-cognitoidp-1-11-868-sources-jar/com/amazonaws/services/cognitoidp/model/transform/AdminDeleteUserFromGroupRequestProtocolMarshaller.java
* /opt/cola/permits/125512452_1642559226.29/0/aws-java-sdk-cognitoidp-1-11-868-sources-jar/com/amazonaws/services/cognitoidp/model/AdminGetUserResult.java
* /opt/cola/permits/125512452_1642559226.29/0/aws-java-sdk-cognitoidp-1-11-868-sources-jar/com/amazonaws/services/cognitoidp/model/transform/ListUsersRequestProtocolMarshaller.java
* /opt/cola/permits/125512452_1642559226.29/0/aws-java-sdk-cognitoidp-1-11-868-sources-jar/com/amazonaws/services/cognitoidp/model/AdminGetUserAttributesRequestMarshaller.java
* /opt/cola/permits/125512452_1642559226.29/0/aws-java-sdk-cognitoidp-1-11-868-sources-jar/com/amazonaws/services/cognitoidp/model/AdminAddUserRequestJava
* /opt/cola/permits/125512452_1642559226.29/0/aws-java-sdk-cognitoidp-1-11-868-sources-jar/com/amazonaws/services/cognitoidp/model/AdminDeleteUserRequestJava
* /opt/cola/permits/125512452_1642559226.29/0/aws-java-sdk-cognitoidp-1-11-868-sources-jar/com/amazonaws/services/cognitoidp/model/AdminInitiateAuthRequestJava
* /opt/cola/permits/125512452_1642559226.29/0/aws-java-sdk-cognitoidp-1-11-868-sources-jar/com/amazonaws/services/cognitoidp/model/AdminForgetDeviceRequestJava
* /opt/cola/permits/125512452_1642559226.29/0/aws-java-sdk-cognitoidp-1-11-868-sources-jar/com/amazonaws/services/cognitoidp/model/AdminSetUserMFAPreferenceRequestJava
* /opt/cola/permits/125512452_1642559226.29/0/aws-java-sdk-cognitoidp-1-11-868-sources-jar/com/amazonaws/services/cognitoidp/model/AdminAddUserRequest
Open Source Used In security_automation identity-guest-cs 753

jar/com/amazonaws/services/cognitoidp/model/RiskLevelType.java
* /opt/cola/permits/125512452_1642559226.29/0/aws-java-sdk-cognitoidp-1-11-868-sources-
jar/com/amazonaws/services/cognitoidp/model/DeleteUserAttributesRequest.java
* /opt/cola/permits/125512452_1642559226.29/0/aws-java-sdk-cognitoidp-1-11-868-sources-
jar/com/amazonaws/services/cognitoidp/model/DeleteUserPoolClientResult.java
* /opt/cola/permits/125512452_1642559226.29/0/aws-java-sdk-cognitoidp-1-11-868-sources-
jar/com/amazonaws/services/cognitoidp/model/CreateUserPoolRequest.java
* /opt/cola/permits/125512452_1642559226.29/0/aws-java-sdk-cognitoidp-1-11-868-sources-
jar/com/amazonaws/services/cognitoidp/model/transform/IdentityProviderTypeMarshaller.java
* /opt/cola/permits/125512452_1642559226.29/0/aws-java-sdk-cognitoidp-1-11-868-sources-
jar/com/amazonaws/services/cognitoidp/model/transform/CreateUserPoolDomainRequestMarshaller.java
* /opt/cola/permits/125512452_1642559226.29/0/aws-java-sdk-cognitoidp-1-11-868-sources-
jar/com/amazonaws/services/cognitoidp/model/transform/SetRiskConfigurationResultJsonUnmarshaller.java
* /opt/cola/permits/125512452_1642559226.29/0/aws-java-sdk-cognitoidp-1-11-868-sources-
jar/com/amazonaws/services/cognitoidp/model/MFAMethodNotFoundException.java
* /opt/cola/permits/125512452_1642559226.29/0/aws-java-sdk-cognitoidp-1-11-868-sources-
jar/com/amazonaws/services/cognitoidp/model/transform/UserPoolTypeJsonUnmarshaller.java
* /opt/cola/permits/125512452_1642559226.29/0/aws-java-sdk-cognitoidp-1-11-868-sources-
jar/com/amazonaws/services/cognitoidp/model/transform/GetCSVHeaderRequestProtocolMarshaller.java
* /opt/cola/permits/125512452_1642559226.29/0/aws-java-sdk-cognitoidp-1-11-868-sources-
jar/com/amazonaws/services/cognitoidp/model/ExplicitAuthFlowsType.java
* /opt/cola/permits/125512452_1642559226.29/0/aws-java-sdk-cognitoidp-1-11-868-sources-
jar/com/amazonaws/services/cognitoidp/model/transform/UpdateUserPoolDomainRequestProtocolMarshaller.java
* /opt/cola/permits/125512452_1642559226.29/0/aws-java-sdk-cognitoidp-1-11-868-sources-
jar/com/amazonaws/services/cognitoidp/model/TokenValidityUnitsType.java
* /opt/cola/permits/125512452_1642559226.29/0/aws-java-sdk-cognitoidp-1-11-868-sources-
jar/com/amazonaws/services/cognitoidp/model/InvalidSmsRoleTrustRelationshipException.java
* /opt/cola/permits/125512452_1642559226.29/0/aws-java-sdk-cognitoidp-1-11-868-sources-
jar/com/amazonaws/services/cognitoidp/model/transform/UICustomizationTypeMarshaller.java
* /opt/cola/permits/125512452_1642559226.29/0/aws-java-sdk-cognitoidp-1-11-868-sources-
jar/com/amazonaws/services/cognitoidp/model/transform/ListDevicesRequestMarshaller.java
* /opt/cola/permits/125512452_1642559226.29/0/aws-java-sdk-cognitoidp-1-11-868-sources-
jar/com/amazonaws/services/cognitoidp/model/transform/UpdateIdentityProviderRequestMarshaller.java
* /opt/cola/permits/125512452_1642559226.29/0/aws-java-sdk-cognitoidp-1-11-868-sources-
jar/com/amazonaws/services/cognitoidp/model/transform/DeviceSecretVerifierConfigTypeMarshaller.java
* /opt/cola/permits/125512452_1642559226.29/0/aws-java-sdk-cognitoidp-1-11-868-sources-
jar/com/amazonaws/services/cognitoidp/model/transform/AnalyticsConfigurationTypeJsonUnmarshaller.java
* /opt/cola/permits/125512452_1642559226.29/0/aws-java-sdk-cognitoidp-1-11-868-sources-
jar/com/amazonaws/services/cognitoidp/model/transform/StopUserImportJobResultJsonUnmarshaller.java
* /opt/cola/permits/125512452_1642559226.29/0/aws-java-sdk-cognitoidp-1-11-868-sources-
jar/com/amazonaws/services/cognitoidp/model/ListTagsForResourceRequest.java
* /opt/cola/permits/125512452_1642559226.29/0/aws-java-sdk-cognitoidp-1-11-868-sources-
jar/com/amazonaws/services/cognitoidp/model/transform/DeviceSecretVerifierConfigTypeMarshaller.java
* /opt/cola/permits/125512452_1642559226.29/0/aws-java-sdk-cognitoidp-1-11-868-sources-
jar/com/amazonaws/services/cognitoidp/model/transform/AnalyticsConfigurationTypeJsonUnmarshaller.java
* /opt/cola/permits/125512452_1642559226.29/0/aws-java-sdk-cognitoidp-1-11-868-sources-
jar/com/amazonaws/services/cognitoidp/model/transform/StopUserImportJobResultJsonUnmarshaller.java
* /opt/cola/permits/125512452_1642559226.29/0/aws-java-sdk-cognitoidp-1-11-868-sources-
jar/com/amazonaws/services/cognitoidp/model/ListTagsForResourceRequest.java
* /opt/cola/permits/125512452_1642559226.29/0/aws-java-sdk-cognitoidp-1-11-868-sources-
1.141 javassist 3.27.0-GA

1.141.1 Available under license:
No license file was found, but licenses were detected in source scan.

/∗
* Javassist, a Java-bytecode translator toolkit.
* Copyright (C) 1999- Shigeru Chiba. All Rights Reserved.
/∗
The contents of this file are subject to the Mozilla Public License Version 1.1 (the "License"); you may not use this file except in compliance with the License. Alternatively, the contents of this file may be used under the terms of the GNU Lesser General Public License Version 2.1 or later, or the Apache License Version 2.0.

Software distributed under the License is distributed on an "AS IS" basis, WITHOUT WARRANTY OF ANY KIND, either express or implied. See the License for the specific language governing rights and limitations under the License.

Found in path(s):
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/tools/rmi/Proxy.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/compiler/ast/Visitor.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/compiler/ast/Member.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/tools/reflect/CannotInvokeException.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/CtNewWrappedConstructor.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/LoaderClassPath.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/tools/rmi/AppletServer.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/ CtConstructor.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/tools/rmi/ObjectNotFoundException.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/compiler/ast/ArrayInit.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/bytecode/Mnemonic.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/bytecode/SignatureAttribute.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/bytecode/ExceptionTable.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/compiler/Parser.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/convert/TransformNewClass.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/compiler/Proxy.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/util/proxy/DefineClassHelper.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/bytecode/analysis/SubroutineScanner.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/ CtMember.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/bytecode/ExceptionTable.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/bytecode/ExceptionTable.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/bytecode/ExceptionTable.java
jar/javassist/expr/ConstructorCall.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1.jar/javassist/tools/rmi/StubGenerator.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1.jar/javassist/bytecode/FieldInfo.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1.jar/javassist/compiler/CodeGen.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1.jar/javassist/bytecode/Analyzer.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1.jar/javassist/bytecode/analysis/IntQueue.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1.jar/javassist/compiler/runtime/DotClass.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1.jar/javassist/Runtime/ProxyObjectInputStream.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1.jar/javassist/bytecode/InstructionPrinter.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1.jar/javassist/bytecode/LocalVariableAttribute.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1.jar/javassist/compiler/ast/CallExpr.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1.jar/javassist/compiler/KeywordTable.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1.jar/javassist/compiler/CodeIterator.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1.jar/javassist/bytecode/analysis/IntQueue.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1.jar/javassist/compiler/KeywordTable.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1.jar/javassist/compiler/CodeIterator.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1.jar/javassist/bytecode/analysis/IntQueue.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1.jar/javassist/compiler/Runtime/Inner.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1.jar/javassist/Runtime/DefinePackageHelper.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1.jar/javassist/bytecode/LineNumberAttribute.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1.jar/javassist/scopedpool/ScopedClassPoolFactoryImpl.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1.jar/javassist/bytecode/LocalVariableAttribute.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1.jar/javassist/compiler/Runtime/Inner.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1.jar/javassist/Runtime/DefinePackageHelper.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1.jar/javassist/bytecode/LineNumberAttribute.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1.jar/javassist/scopedpool/ScopedClassPoolFactory.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1.jar/javassist/Modifier.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1.jar/javassist/bytecode/annotation/AnnotationsWriter.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1.jar/javassist/bytecode/analysis/Subroutine.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1.jar/javassist/compiler/Runtime/Inner.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1.jar/javassist/Runtime/DefinePackageHelper.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1.jar/javassist/bytecode/LineNumberAttribute.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1.jar/javassist/scopedpool/ScopedClassPoolFactory.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1.jar/javassist/Modifier.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1.jar/javassist/bytecode/annotation/AnnotationsWriter.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-
Open Source Used in security_automation identity-guest-cs

* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/bytecode/StackMap.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/NotFoundException.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/tools/reflect/Metalevel.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/compiler/ast/Variable.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/tools/reflect/Sample.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/Instanceof.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/compiler/ast/Stmnt.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/tools/framedump.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/compiler/ast/Expr.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/Bytecode.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/bytecode/ConstantAttribute.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/runtime/Desc.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/compiler/ast/Declarator.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/Declarator.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/javaassist/ProxyFactory.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/tools/rmi/RemoteException.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/ast/InstanceOfExpr.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/ast/Declator.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/javaassist/stackmap/TypedBlock.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/tools/rmi/ObjectImporter.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/expr/ExprEditor.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/compiler/ProceedHandler.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/bytecode/ExceptionsAttribute.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/compiler/ast/MethodDecl.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/bytecode/InnerClassesAttribute.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/compiler/ast/Symbol.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/tools/web/Viewer.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/ByteArrayClassPath.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/compiler/AccessorMaker.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/compiler/Lex.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/compiler/ast/ASTList.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/tools/reflect/Loader.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/CtField.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/tools/Dump.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/compiler/CompileError.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/compiler/ExprHandler.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/compiler/ByteStream.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/compiler/stackmap/Tracer.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/util/HotSwapAgent.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/compiler/FieldAccess.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/compiler/Bytecode/InnerClassesAttribute.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/tools/web/Viewer.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/tools/util/proxy/ProxyObject.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/compiler/TransformCall.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/compiler/ast/Symbol.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/compiler/ByteArrayListPath.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/compiler/AccessorMaker.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/compiler/Lex.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/compiler/ast/ASTList.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/expr/MethodCall.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/CtField.java
No license file was found, but licenses were detected in source scan.

/*

* Javassist, a Java-bytecode translator toolkit.
* Copyright (C) 2004 Bill Burke. All Rights Reserved.
* The contents of this file are subject to the Mozilla Public License Version
* 1.1 (the "License"); you may not use this file except in compliance with
* the License. Alternatively, the contents of this file may be used under
* the terms of the GNU Lesser General Public License Version 2.1 or later,
* or the Apache License Version 2.0.
* Software distributed under the License is distributed on an "AS IS" basis,
* WITHOUT WARRANTY OF ANY KIND, either express or implied. See the License
* for the specific language governing rights and limitations under the
* License.
*/
1.142 consul-api 1.4.5

1.142.1 Available under license:
Apache-2.0

1.143 netty-buffer 4.1.69.Final

1.143.1 Available under license:
No license file was found, but licenses were detected in source scan.

# The Netty Project licenses this file to you under the Apache License,
# version 2.0 (the "License"); you may not use this file except in compliance
# with the License. You may obtain a copy of the License at:
# distributed under the License is distributed on an "AS IS" BASIS, WITHOUT

Found in path(s):
* /opt/cola/permits/1265673620_1643932634.48/0/netty-buffer-4-1-69-final-sources-jar/META-INF/native-
image/io.netty/buffer/native-image.properties
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2012 The Netty Project
 *
 * The Netty Project licenses this file to the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1265673620_1643932634.48/0/netty-buffer-4-1-69-final-sources-
jar/io/netty/buffer/PooledHeapByteBuf.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2020 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1265673620_1643932634.48/0/netty-buffer-4-1-69-final-sources-
jar/io/netty/buffer/LongPriorityQueue.java
* /opt/cola/permits/1265673620_1643932634.48/0/netty-buffer-4-1-69-final-sources-
jar/io/netty/buffer/SizeClasses.java
* /opt/cola/permits/1265673620_1643932634.48/0/netty-buffer-4-1-69-final-sources-jar/io/netty/buffer/search/package-info.java
* /opt/cola/permits/1265673620_1643932634.48/0/netty-buffer-4-1-69-final-sources-jar/io/netty/buffer/SizeClassesMetric.java
* /opt/cola/permits/1265673620_1643932634.48/0/netty-buffer-4-1-69-final-sources-jar/io/netty/buffer/LongLongHashMap.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015 The Netty Project
 * *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * *
 *  https://www.apache.org/licenses/LICENSE-2.0
 * *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1265673620_1643932634.48/0/netty-buffer-4-1-69-final-sources-jar/io/netty/buffer/UnpooledUnsafeHeapByteBuf.java
* /opt/cola/permits/1265673620_1643932634.48/0/netty-buffer-4-1-69-final-sources-jar/io/netty/buffer/PooledUnsafeHeapByteBuf.java
* /opt/cola/permits/1265673620_1643932634.48/0/netty-buffer-4-1-69-final-sources-jar/io/netty/buffer/PoolSubpageMetric.java
* /opt/cola/permits/1265673620_1643932634.48/0/netty-buffer-4-1-69-final-sources-jar/io/netty/buffer/PoolArenaMetric.java
* /opt/cola/permits/1265673620_1643932634.48/0/netty-buffer-4-1-69-final-sources-jar/io/netty/buffer/UnsafeByteBufUtil.java
* /opt/cola/permits/1265673620_1643932634.48/0/netty-buffer-4-1-69-final-sources-jar/io/netty/buffer/PoolChunkMetric.java
* /opt/cola/permits/1265673620_1643932634.48/0/netty-buffer-4-1-69-final-sources-jar/io/netty/buffer/UnpooledSlicedByteBuf.java
* /opt/cola/permits/1265673620_1643932634.48/0/netty-buffer-4-1-69-final-sources-jar/io/netty/buffer/AbstractUnsafeSwappedByteBuf.java
* /opt/cola/permits/1265673620_1643932634.48/0/netty-buffer-4-1-69-final-sources-jar/io/netty/buffer/PoolChunkListMetric.java
* /opt/cola/permits/1265673620_1643932634.48/0/netty-buffer-4-1-69-final-sources-jar/io/netty/buffer/UnpooledDuplicatedByteBuf.java

Open Source Used In security_automation identity-guest-cs 777
No license file was found, but licenses were detected in source scan.

~ Copyright 2012 The Netty Project
~
~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~
~ https://www.apache.org/licenses/LICENSE-2.0
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.

Found in path(s):
* /opt/cola/permits/1265673620_1643932634.48/0/netty-buffer-4-1-69-final-sources-jar/META-INF/maven/io.netty/netty-buffer/pom.xml

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2016 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1265673620_1643932634.48/0/netty-buffer-4-1-69-final-sources-jar/io/netty/buffer/UnpooledUnsafeNoCleanerDirectByteBuf.java
* /opt/cola/permits/1265673620_1643932634.48/0/netty-buffer-4-1-69-final-sources-jar/io/netty/buffer/AdvancedLeakAwareCompositeByteBuf.java
* /opt/cola/permits/1265673620_1643932634.48/0/netty-buffer-4-1-69-final-sources-jar/io/netty/buffer/AbstractPooledDerivedByteBuf.java
* /opt/cola/permits/1265673620_1643932634.48/0/netty-buffer-4-1-69-final-sources-
/*
* Copyright 2013 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

No license file was found, but licenses were detected in source scan.

Found in path(s):
* /opt/cola/permits/1265673620_1643932634.48/netty-buffer-4-1-69-final-sources-jar/io/netty/buffer/WrappedByteBuf.java
* /opt/cola/permits/1265673620_1643932634.48/netty-buffer-4-1-69-final-sources-jar/io/netty/buffer/EmptyByteBuf.java
* /opt/cola/permits/1265673620_1643932634.48/netty-buffer-4-1-69-final-sources-jar/io/netty/buffer/PooledUnsafeDirectByteBuf.java
* /opt/cola/permits/1265673620_1643932634.48/netty-buffer-4-1-69-final-sources-jar/io/netty/buffer/AbstractDerivedByteBuf.java
* /opt/cola/permits/1265673620_1643932634.48/netty-buffer-4-1-69-final-sources-jar/io/netty/buffer/DefaultByteBufHolder.java
* /opt/cola/permits/1265673620_1643932634.48/netty-buffer-4-1-69-final-sources-jar/io/netty/buffer/ReadOnlyUnsafeDirectByteBuf.java
* /opt/cola/permits/1265673620_1643932634.48/netty-buffer-4-1-69-final-sources-jar/io/netty/buffer/ReadOnlyByteBufferBuf.java
* /opt/cola/permits/1265673620_1643932634.48/netty-buffer-4-1-69-final-sources-jar/io/netty/buffer/FixedCompositeByteBuf.java

/*
* Copyright 2020 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License, version 2.0 (the
* "License"); you may not use this file except in compliance with the License. You may obtain a
* copy of the License at:
*
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

No license file was found, but licenses were detected in source scan.

/*
* Found in path(s):
* */

No license file was found, but licenses were detected in source scan.
* Copyright 2017 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* 
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1265673620_1643932634.48/0/netty-buffer-4-1-69-final-sources-
  jar/io/netty/buffer/ByteBufAllocatorMetric.java
* /opt/cola/permits/1265673620_1643932634.48/0/netty-buffer-4-1-69-final-sources-
  jar/io/netty/buffer/PooledByteBufAllocatorMetric.java
* /opt/cola/permits/1265673620_1643932634.48/0/netty-buffer-4-1-69-final-sources-
  jar/io/netty/buffer/ByteBufAllocatorMetricProvider.java

No license file was found, but licenses were detected in source scan.

*/

* Copyright 2014 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* 
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1265673620_1643932634.48/0/netty-buffer-4-1-69-final-sources-
  jar/io/netty/buffer/UnsafeHeapSwappedByteBuf.java
* /opt/cola/permits/1265673620_1643932634.48/0/netty-buffer-4-1-69-final-sources-
  jar/io/netty/buffer/UnsafeDirectSwappedByteBuf.java

No license file was found, but licenses were detected in source scan.

/]
* Copyright 2012 The Netty Project

* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* https://www.apache.org/licenses/LICENSE-2.0

* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1265673620_1643932634.480/netty-buffer-4-1-69-final-sources-
  jar/io/netty/buffer/Unpooled.java
* /opt/cola/permits/1265673620_1643932634.480/netty-buffer-4-1-69-final-sources-
  jar/io/netty/buffer/UnpooledUnsafeDirectByteBuf.java
* /opt/cola/permits/1265673620_1643932634.480/netty-buffer-4-1-69-final-sources-
  jar/io/netty/buffer/PooledDirectByteBuf.java
* /opt/cola/permits/1265673620_1643932634.480/netty-buffer-4-1-69-final-sources-
  jar/io/netty/buffer/PooledByteBuf.java
* /opt/cola/permits/1265673620_1643932634.480/netty-buffer-4-1-69-final-sources-
  jar/io/netty/buffer/ByteBufUtil.java
* /opt/cola/permits/1265673620_1643932634.480/netty-buffer-4-1-69-final-sources-
  jar/io/netty/buffer/PoolChunk.java
* /opt/cola/permits/1265673620_1643932634.480/netty-buffer-4-1-69-final-sources-
  jar/io/netty/buffer/AbstractByteBuf.java
* /opt/cola/permits/1265673620_1643932634.480/netty-buffer-4-1-69-final-sources-
  jar/io/netty/buffer/PooledByteBufAllocator.java
* /opt/cola/permits/1265673620_1643932634.480/netty-buffer-4-1-69-final-sources-
  jar/io/netty/buffer/PoolThreadCache.java
* /opt/cola/permits/1265673620_1643932634.480/netty-buffer-4-1-69-final-sources-
  jar/io/netty/buffer/AbstractByteBufAllocator.java
* /opt/cola/permits/1265673620_1643932634.480/netty-buffer-4-1-69-final-sources-
  jar/io/netty/buffer/PoolSubpage.java
* /opt/cola/permits/1265673620_1643932634.480/netty-buffer-4-1-69-final-sources.jar/
  io/netty/buffer/package-info.java
* /opt/cola/permits/1265673620_1643932634.480/netty-buffer-4-1-69-final-sources-
  jar/io/netty/buffer/UnpooledByteBufAllocator.java
* /opt/cola/permits/1265673620_1643932634.480/netty-buffer-4-1-69-final-sources-
  jar/io/netty/buffer/UnpooledDirectByteBuf.java
* /opt/cola/permits/1265673620_1643932634.480/netty-buffer-4-1-69-final-sources-
  jar/io/netty/buffer/CompositeByteBuf.java
* /opt/cola/permits/1265673620_1643932634.480/netty-buffer-4-1-69-final-sources-
  jar/io/netty/buffer/UnpooledHeapByteBuf.java
1.144 jakarta-annotation-api 1.3.5
1.144.1 Available under license:

# Eclipse Public License - v 2.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial content
   Distributed under this Agreement, and

b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;
where such changes and/or additions to the Program originate from and are Distributed by that particular Contributor. A Contribution "originates" from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include changes or additions to the Program that are not Modified Works.
"Contributor" means any person or entity that Distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions Distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement or any Secondary License (as applicable), including Contributors.

"Derivative Works" shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

"Modified Works" shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

"Distribute" means the acts of a) distributing or b) making available in any manner that enables the transfer of a copy.

"Source Code" means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Secondary License" means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent
license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

e) Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. REQUIREMENTS

3.1 If a Contributor Distributes the Program in any form, then:

a) the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

b) the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:
   i) effectively disclaims on behalf of all other Contributors all
warranties and conditions, express and implied, including
warranties or conditions of title and non-infringement, and
implied warranties or conditions of merchantability and fitness
for a particular purpose;

ii) effectively excludes on behalf of all other Contributors all
liability for damages, including direct, indirect, special,
incidental and consequential damages, such as lost profits;

iii) does not attempt to limit or alter the recipients’ rights
in the Source Code under section 3.2; and

iv) requires any subsequent distribution of the Program by any
party to be under a license that satisfies the requirements
of this section 3.

3.2 When the Program is Distributed as Source Code:

a) it must be made available under this Agreement, or if the
Program (i) is combined with other material in a separate file or
files made available under a Secondary License, and (ii) the initial
Contributor attached to the Source Code the notice described in
Exhibit A of this Agreement, then the Program may be made available
under the terms of such Secondary Licenses, and

b) a copy of this Agreement must be included with each copy of
the Program.

3.3 Contributors may not remove or alter any copyright, patent,
trademark, attribution notices, disclaimers of warranty, or limitations
of liability ("notices") contained within the Program from any copy of
the Program which they Distribute, provided that Contributors may add
their own appropriate notices.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities
with respect to end users, business partners and the like. While this
license is intended to facilitate the commercial use of the Program,
the Contributor who includes the Program in a commercial product
offering should do so in a manner which does not create potential
liability for other Contributors. Therefore, if a Contributor includes
the Program in a commercial product offering, such Contributor
("Commercial Contributor") hereby agrees to defend and indemnify every
other Contributor ("Indemnified Contributor") against any losses,
damages and costs (collectively "Losses") arising from claims, lawsuits
and other legal actions brought by a third party against the Indemnified
Contributor to the extent caused by the acts or omissions of such
Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL
If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice
"This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}.

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

---

## The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor
Boston, MA 02110-1335
USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you
distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously
and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program
with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in
full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.
8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED
INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF
THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR
OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest
possible use to the public, the best way to achieve this is to make it
free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to
attach them to the start of each source file to most effectively convey
the exclusion of warranty; and each file should have at least the
"copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful, but
WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program; if not, write to the Free Software
Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1335 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this
when it starts in an interactive mode:

    Gnomovision version 69, Copyright (C) year name of author
    Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type
    `show w'. This is free software, and you are welcome to redistribute
    it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the
appropriate parts of the General Public License. Of course, the commands
you use may be called something other than `show w' and `show c'; they
could even be mouse-clicks or menu items--whatever suits your program.
You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

---

## CLASSPATH EXCEPTION

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

# Notices for Jakarta Annotations

This content is produced and maintained by the Jakarta Annotations project.

* Project home: https://projects.eclipse.org/projects/ee4j.ca

## Trademarks

Jakarta Annotations is a trademark of the Eclipse Foundation.
## Declared Project Licenses

This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0 which is available at http://www.eclipse.org/legal/epl-2.0. This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License v. 2.0 are satisfied: GNU General Public License, version 2 with the GNU Classpath Exception which is available at https://www.gnu.org/software/classpath/license.html.

SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

## Source Code

The project maintains the following source code repositories:

* https://github.com/eclipse-ee4j/common-annotations-api

## Third-party Content

## Cryptography

Content may contain encryption software. The country in which you are currently may have restrictions on the import, possession, and use, and/or re-export to another country, of encryption software. BEFORE using any encryption software, please check the country's laws, regulations and policies concerning the import, possession, or use, and re-export of encryption software, to see if this is permitted.

### 1.145 esri-geometry-api 1.2.1

#### 1.145.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*
Copyright 1995-2015 Esri

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

For additional information, contact:
Environmental Systems Research Institute, Inc.
Attn: Contracts Dept
380 New York Street
Redlands, California, USA 92373

e-mail: contracts@esri.com
*/

Found in path(s):
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/BucketSort.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/Point.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/OperatorShapePreservingDensify.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/Envelope3D.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/ProgressTracker.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/PolygonUtils.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/OperatorBoundary.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/AttributeStreamOfDbl.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/GeometryException.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/Simplificator.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/MultiVertexGeometry.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/OperatorCutCursor.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/OperatorInternalRelationUtils.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/OperatorProject.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/TopoGraph.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/OperatorImportFromGeoJsonLocal.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/OperatorImportFromWkbLocal.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/OperatorImportFromWkbLocal.java
Open Source Used in security_automation identity-guest-cs

* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/GeometryEngine.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/JSONObjectEnumerator.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/MathUtils.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/OperatorDistance.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/OperatorDisjointLocal.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/MultiPoint.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/PlaneSweepCrackerHelper.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/OperatorExportToESRIShapeCursor.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/WkbByteOrder.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/TopologicalOperations.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/OperatorExportToJsonCursor.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1.jar/com/esri/core/geometry/QuadTreeImpl.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1.jar/com/esri/core/geometry/OperatorOffsetLocal.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1.jar/com/esri/core/geometry/ShapeModifiers.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1.jar/com/esri/core/geometry/OperatorCut.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1.jar/com/esri/core/geometry/ConvexHull.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1.jar/com/esri/core/geometry/OperatorGeneralizeCursor.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1.jar/com/esri/core/geometry/JsonReader.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1.jar/com/esri/core/geometry/Envelope1D.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1.jar/com/esri/core/geometry/ListeningGeometryCursor.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1.jar/com/esri/core/geometry/MapGeometryCursor.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1.jar/com/esri/core/geometry/WkbImportFlags.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1.jar/com/esri/core/geometry/OperatorSimplifyLocal.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1.jar/com/esri/core/geometry/RelationalOperationsMatrix.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1.jar/com/esri/core/geometry/OperatorImportFromJsonLocal.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1.jar/com/esri/core/geometry/OperatorExportToESRIShapeLocal.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1.jar/com/esri/core/geometry/OperatorDifference.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1.jar/com/esri/core/geometry/UserCancelException.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1.jar/com/esri/core/geometry/ShapeExportFlags.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1.jar/com/esri/core/geometry/OperatorGeodeticDensifyByLength.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1.jar/com/esri/core/geometry/StridedIndexTypeCollection.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1.jar/com/esri/core/geometry/OperatorClipLocal.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1.jar/com/esri/core/geometry/SegmentBuffer.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1.jar/com/esri/core/geometry/MgrsConversionMode.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1.jar/com/esri/core/geometry/JsonParserCursor.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/OperatorCutLocal.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/OperatorImportFromWkt.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/OperatorExportToWkbLocal.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/OperatorImportFromESRIShapeLocal.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/SpatialReferenceSerializer.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/Clusterer.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/Polygon.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/OperatorDensifyByLength.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/OperatorIntersectsLocal.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/PathFlags.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/AttributeStreamOfInt16.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/OperatorProjectLocal.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/PathFlags.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/Line.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/OperatorIntersectsLocal.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/OperatorWithinLocal.java

---

Open Source Used In security_automation identity-guest-cs 803
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1.jar/com/esri/core/geometry/OperatorSimplifyCursorOGC.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1.jar/com/esri/core/geometry/OperatorBoundaryLocalCursor.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1.jar/com/esri/core/geometry/OperatorGeneralize.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1.jar/com/esri/core/geometry/StringUtils.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1.jar/com/esri/core/geometry/OperatorExportToWkt.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1.jar/com/esri/core/geometry/OperatorIntersectCursorOGC.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1.jar/com/esri/core/geometry/OperatorIntersection.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1.jar/com/esri/core/geometry/OperatorIntersects.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1.jar/com/esri/core/geometry/ShapeImportFlags.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1.jar/com/esri/core/geometry/Proximity2DResult.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1.jar/com/esri/core/geometry/SpatialReference.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1.jar/com/esri/core/geometry/Treap.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1.jar/com/esri/core/geometry/Transformation2D.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1.jar/com/esri/core/geometry/OperatorGeodesicBuffer.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1.jar/com/esri/core/geometry/Polyline.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1.jar/com/esri/core/geometry/JsonParserReader.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1.jar/com/esri/core/geometry/OperatorImportFromJsonCursor.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1.jar/com/esri/core/geometry/OperatorUnionCursor.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1.jar/com/esri/core/geometry/OperatorSymmetricDifferenceLocal.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1.jar/com/esri/core/geometry/WkbExportFlags.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1.jar/com/esri/core/geometry/OperatorShapePreservingDensifyLocal.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1.jar/com/esri/core/geometry/Bufferer.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1.jar/com/esri/core/geometry/MapOGCSStructure.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1.jar/com/esri/core/geometry/OperatorExportToJson.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1.jar/com/esri/core/geometry/VertexDescription.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/OperatorDisjoint.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/OperatorSimpleRelation.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/Boundary.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/ProjectionTransformation.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/GeoDist.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/IntervalTreeImpl.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/OperatorRelateLocal.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/OGCStructure.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/GeometryAccelerators.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/JsonStringWriter.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/OperatorExportToJsonLocal.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/OperatorOverlapsLocal.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/OperatorIntersectionCursor.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/Geometry.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/DirtyFlags.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/JsonCursor.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/MultiPointImpl.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/OperatorImportFromESRIShapeCursor.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/OperatorCrossesLocal.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/Envelope.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/Proximity2DResultComparator.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/PolylinePath.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/OperatorDifferenceLocal.java
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-jar/com/esri/core/geometry/Point3D.java
No license file was found, but licenses were detected in source scan.

/*
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

For additional information, contact:
Environmental Systems Research Institute, Inc.
Attn: Contracts Dept
380 New York Street
Redlands, California, USA 92373

e-mail: contracts@esri.com
*/

Found in path(s):
* /opt/cola/permits/1199743771_1630577635.90/esri-geometry-api-1-2-1-sources-1-
jar/com/esri/core/geometry/OperatorExportToGeoJsonLocal.java
* /opt/cola/permits/1199743771_1630577635.90/esri-geometry-api-1-2-1-sources-1-
jar/com/esri/core/geometry/OperatorExportToGeoJson.java
* /opt/cola/permits/1199743771_1630577635.90/esri-geometry-api-1-2-1-sources-1-
jar/com/esri/core/geometry/OperatorExportToGeoJsonCursor.java
No license file was found, but licenses were detected in source scan.

/*
Copyright 1995-2013 Esri

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at
http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

For additional information, contact:
Environmental Systems Research Institute, Inc.
Attn: Contracts Dept
380 New York Street
Redlands, California, USA 92373
email: contracts@esri.com
*/

Found in path(s):
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-
jar/com/esri/core/geometry/CombineOperator.java
No license file was found, but licenses were detected in source scan.

/*
Copyright 2013-2015 Esri

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

For additional information, contact:
Environmental Systems Research Institute, Inc.
Attn: Contracts Dept
380 New York Street
Redlands, California, USA 92373
email: contracts@esri.com
*/

Found in path(s):
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-
jar/com/esri/core/geometry/SimpleRasterizer.java
No license file was found, but licenses were detected in source scan.

/*
Copyright 1995-2013 Esri

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

For additional information, contact:
Environmental Systems Research Institute, Inc.
Attn: Contracts Dept
380 New York Street
Redlands, California, USA 92373

email: contracts@esri.com
*/

Found in path(s):
* /opt/cola/permits/1199743771_1630577635.9/0/esri-geometry-api-1-2-1-sources-1-
jar/com/esri/core/geometry/RasterizedGeometry2DImpl.java

1.146 content-type 2.1
1.146.1 Available under license:
No license file was found, but licenses were detected in source scan.

/*
* oauth2-oidc-sdk
*
* Copyright 2020, Connect2id Ltd and contributors.
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use
* this file except in compliance with the License. You may obtain a copy of the
* License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed
* under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR
* CONDITIONS OF ANY KIND, either express or implied. See the License for the
* specific language governing permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1143230880_1615547501.65/0/content-type-2-1-sources-
  jar/com/nimbusds/common/contenttype/ContentType.java
No license file was found, but licenses were detected in source scan.


Found in path(s):
* /opt/cola/permits/1143230880_1615547501.65/0/content-type-2-1-sources-jar/META-
  INF/maven/com.nimbusds/content-type/pom.xml

1.147 snappy-java 1.1.7.7

1.147.1 Available under license:
This product includes software developed by Google
Snappy: http://code.google.com/p/snappy/ (New BSD License)

This product includes software developed by Apache
(Apache 2.0 license)

This library containd statically linked libstdc++. This inclusion is allowed by
"GCC RUNtime Library Exception"

== Contributors ==
* Tatu Saloranta
* Providing benchmark suite
* Alec Wysoker
* Performance and memory usage improvement

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.
"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
1.148 jackson-datatype-joda 2.13.0

1.148.1 Available under license:
This copy of Jackson JSON processor streaming parser/generator is licensed under the Apache (Software) License, version 2.0 ("the License"). See the License for details about distribution rights, and the specific rights regarding derivate works.

You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

1.149 commons-logging 1.2

1.149.1 Available under license:
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2001-2006 The Apache Software Foundation.
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2001-2004,2006 The Apache Software Foundation.
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* /opt/cola/permits/1135840457_1613613080.13/0/commons-logging-1.1-sources-9.jar/org/apache/commons/logging/LogFactory.java
*/
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1135840457_1613613080.13/0/commons-logging-1-1-sources-9-
  jar/org/apache/commons/logging/impl/AvalonLogger.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2004 The Apache Software Foundation.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1135840457_1613613080.13/0/commons-logging-1-1-sources-9-
  jar/org/apache/commons/logging/impl/WeakHashtable.java
No license file was found, but licenses were detected in source scan.

2004 The Apache Software Foundation.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Found in path(s):
* /opt/cola/permits/1135840457_1613613080.13/0/commons-logging-1-1-sources-9-jar/org/apache/commons/logging/package.html
* /opt/cola/permits/1135840457_1613613080.13/0/commons-logging-1-1-sources-9-jar/org/apache/commons/logging/impl/package.html

No license file was found, but licenses were detected in source scan.

/*@*/

* Copyright 2001-2004 The Apache Software Foundation.
* *
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* *
* http://www.apache.org/licenses/LICENSE-2.0
* *
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1135840457_1613613080.13/0/commons-logging-1-1-sources-9-jar/org/apache/commons/logging/LogConfigurationException.java
* /opt/cola/permits/1135840457_1613613080.13/0/commons-logging-1-1-sources-9-jar/org/apache/commons/logging/impl/Log4JLogger.java
* /opt/cola/permits/1135840457_1613613080.13/0/commons-logging-1-1-sources-9-jar/org/apache/commons/logging/impl/LogFactoryImpl.java
* /opt/cola/permits/1135840457_1613613080.13/0/commons-logging-1-1-sources-9-jar/org/apache/commons/logging/impl/Jdk13LumberjackLogger.java
* /opt/cola/permits/1135840457_1613613080.13/0/commons-logging-1-1-sources-9-jar/org/apache/commons/logging/impl/SimpleLog.java
* /opt/cola/permits/1135840457_1613613080.13/0/commons-logging-1-1-sources-9-jar/org/apache/commons/logging/impl/Jdk14Logger.java
* /opt/cola/permits/1135840457_1613613080.13/0/commons-logging-1-1-sources-9-jar/org/apache/commons/logging/LogSource.java
* /opt/cola/permits/1135840457_1613613080.13/0/commons-logging-1-1-sources-9-jar/org/apache/commons/logging/impl/NoOpLog.java
* /opt/cola/permits/1135840457_1613613080.13/0/commons-logging-1-1-sources-9-jar/org/apache/commons/logging/impl/LogKitLogger.java
* /opt/cola/permits/1135840457_1613613080.13/0/commons-logging-1-1-sources-9-jar/org/apache/commons/logging/Log.java

No license file was found, but licenses were detected in source scan.
Found in path(s):
* /opt/cola/permits/1135840457_1613613080.13/0/commons-logging-1-1-sources-9-
  jar/org/apache/commons/logging/impl/ServletContextCleaner.java

1.150 activation 1.2.0
1.150.1 Available under license:
COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.1

1. Definitions.

1.1. "Contributor" means each individual or entity that creates or
     contributes to the creation of Modifications.

1.2. "Contributor Version" means the combination of the Original
     Software, prior Modifications used by a Contributor (if any), and
     the Modifications made by that particular Contributor.

1.3. "Covered Software" means (a) the Original Software, or (b)
     Modifications, or (c) the combination of files containing Original
     Software with files containing Modifications, in each case including
     portions thereof.

1.4. "Executable" means the Covered Software in any form other than
     Source Code.

1.5. "Initial Developer" means the individual or entity that first
     makes Original Software available under this License.

1.6. "Larger Work" means a work which combines Covered Software or
     portions thereof with code not governed by the terms of this License.
1.7. "License" means this document.

1.8. "Licensable" means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. "Modifications" means the Source Code and Executable form of any of the following:

A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

B. Any new file that contains any part of the Original Software or previous Modification; or

C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. "Original Software" means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. "Patent Claims" means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. "Source Code" means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. "You" (or "Your") means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, "You" includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer
hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the
Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients’ rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software.
 However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.


You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.

Oracle is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software
prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as "Participant") alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or
indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. If You assert a patent infringement claim against Participant alleging that the Participant Software directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

6.4. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a "commercial item," as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of "commercial computer software" (as that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and "commercial computer software documentation" as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent
with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

-------------------------------

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)

The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the
state courts of the State of California, with venue lying in Santa Clara County, California.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor
Boston, MA 02110-1335
USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain
that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:
a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the
Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may
differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.
To do so, attach the following notices to the program. It is safest to
attach them to the start of each source file to most effectively convey
the exclusion of warranty; and each file should have at least the
"copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful, but
WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program; if not, write to the Free Software
Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1335 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this
when it starts in an interactive mode:

    Gnomovision version 69, Copyright (C) year name of author
    Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type
    `show w'. This is free software, and you are welcome to redistribute
    it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the
appropriate parts of the General Public License. Of course, the commands
you use may be called something other than `show w' and `show c'; they
could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the program, if
necessary. Here is a sample; alter the names:

    Yoyodyne, Inc., hereby disclaims all copyright interest in the
    program `Gnomovision' (which makes passes at compilers) written by
    James Hacker.

    signature of Ty Coon, 1 April 1989
    Ty Coon, President of Vice
This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

#

Certain source files distributed by Oracle America, Inc. and/or its affiliates are subject to the following clarification and special exception to the GPLv2, based on the GNU Project exception for its Classpath libraries, known as the GNU Classpath Exception, but only where Oracle has expressly included in the particular source file’s header the words "Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the LICENSE file that accompanied this code."

You should also note that Oracle includes multiple, independent programs in this software package. Some of those programs are provided under licenses deemed incompatible with the GPLv2 by the Free Software Foundation and others. For example, the package includes programs licensed under the Apache License, Version 2.0. Such programs are licensed to you under their original licenses.

Oracle facilitates your further distribution of this package by adding the Classpath Exception to the necessary parts of its GPLv2 code, which permits you to use that code in combination with other independent modules not licensed under the GPLv2. However, note that this would not permit you to commingle code under an incompatible license with Oracle's GPLv2 licensed code by, for example, cutting and pasting such code into a file also containing Oracle's GPLv2 licensed code and then distributing the result. Additionally, if you were to remove the Classpath Exception from any of the files to which it applies and distribute the result, you would likely be required to license some or all of the other code in that distribution under the GPLv2 as well, and since the GPLv2 is incompatible with the license terms of some items included in the distribution by Oracle, removing the Classpath Exception could therefore effectively compromise your ability to further distribute the package.

Proceed with caution and we recommend that you obtain the advice of a lawyer skilled in open source matters before removing the Classpath Exception or making modifications to this package which may subsequently be redistributed and/or involve the use of third party software.

CLASSPATH EXCEPTION
Linking this library statically or dynamically with other modules is
making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

1.151 zstd 1.4.4
1.151.1 Available under license :
BSD License

For Zstandard software

Copyright (c) 2016-present, Facebook, Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name Facebook nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
License is intended to guarantee your freedom to share and change free
software--to make sure the software is free for all its users. This
General Public License applies to most of the Free Software
Foundation's software and to any other program whose authors commit to
using it. (Some other Free Software Foundation software is covered by
the GNU Lesser General Public License instead.) You can apply it to
your programs, too.

When we speak of free software, we are referring to freedom, not
price. Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for
this service if you wish), that you receive source code or can get it
if you want it, that you can change the software or use pieces of it
in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid
anyone to deny you these rights or to ask you to surrender the rights.
These restrictions translate to certain responsibilities for you if you
distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether
gratis or for a fee, you must give the recipients all the rights that
you have. You must make sure that they, too, receive or can get the
source code. And you must show them these terms so they know their
rights.

We protect your rights with two steps: (1) copyright the software, and
(2) offer you this license which gives you legal permission to copy,
distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain
that everyone understands that there is no warranty for this free
software. If the software is modified by someone else and passed on, we
want its recipients to know that what they have is not the original, so

(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991
that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:
a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   b) Accompany it with a written offer, valid for at least three
years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the
original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will
be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.
To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice
This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License.

1.152 spring-cloud-consul-config 3.0.4

1.152.1 Available under license:
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2013-2020 the original author or authors.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 *   https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1222421590_1635579093.08/0/spring-cloud-consul-config-3-0-4-sources-jar/org/springframework/cloud/consul/config/ConsulRetryBootstrapper.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015-2021 the original author or authors.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 *   https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
Found in path(s):
* /opt/cola/permits/1222421590_1635579093.08/0/spring-cloud-consul-config-3-0-4-sources-jar/org/springframework/cloud/consul/config/ConsulConfigDataMissingEnvironmentPostProcessor.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015-2020 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1222421590_1635579093.08/0/spring-cloud-consul-config-3-0-4-sources-jar/org/springframework/cloud/consul/config/ConsulPropertySources.java
* /opt/cola/permits/1222421590_1635579093.08/0/spring-cloud-consul-config-3-0-4-sources-jar/org/springframework/cloud/consul/config/ConsulConfigDataResource.java
* /opt/cola/permits/1222421590_1635579093.08/0/spring-cloud-consul-config-3-0-4-sources-jar/org/springframework/cloud/consul/config/ConsulBootstrapper.java
* /opt/cola/permits/1222421590_1635579093.08/0/spring-cloud-consul-config-3-0-4-sources-jar/org/springframework/cloud/consul/config/ConsulConfigIndexes.java
* /opt/cola/permits/1222421590_1635579093.08/0/spring-cloud-consul-config-3-0-4-sources-jar/org/springframework/cloud/consul/config/ConsulConfigDataLoader.java
* /opt/cola/permits/1222421590_1635579093.08/0/spring-cloud-consul-config-3-0-4-sources-jar/org/springframework/cloud/consul/config/ConsulConfigDataLocationResolver.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2013-2019 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1222421590_1635579093.08/0/spring-cloud-consul-config-3-0-4-sources-jar/org/springframework/cloud/consul/config/ConsulConfigAutoConfiguration.java
* /opt/cola/permits/1222421590_1635579093.08/0/spring-cloud-consul-config-3-0-4-sources-jar/org/springframework/cloud/consul/config/ConsulFilesPropertySource.java
* /opt/cola/permits/1222421590_1635579093.08/0/spring-cloud-consul-config-3-0-4-sources-jar/org/springframework/cloud/consul/config/ConsulConfigBootstrapConfiguration.java
* /opt/cola/permits/1222421590_1635579093.08/0/spring-cloud-consul-config-3-0-4-sources-jar/org/springframework/cloud/consul/config/ConsulConfigProperties.java
* /opt/cola/permits/1222421590_1635579093.08/0/spring-cloud-consul-config-3-0-4-sources-jar/org/springframework/cloud/consul/config/PropertySourcesLocatedEvent.java
* /opt/cola/permits/1222421590_1635579093.08/0/spring-cloud-consul-config-3-0-4-sources-jar/org/springframework/cloud/consul/config/ConsulPropertySource.java
* /opt/cola/permits/1222421590_1635579093.08/0/spring-cloud-consul-config-3-0-4-sources-jar/org/springframework/cloud/consul/config/ConfigWatch.java
* /opt/cola/permits/1222421590_1635579093.08/0/spring-cloud-consul-config-3-0-4-sources-jar/org/springframework/cloud/consul/config/ConsulPropertySourceLocator.java

No license file was found, but licenses were detected in source scan.

<name>Apache License, Version 2.0</name>
Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at
distributed under the License is distributed on an "AS IS" BASIS.

Found in path(s):
* /opt/cola/permits/1222421590_1635579093.08/0/spring-cloud-consul-config-3-0-4-sources-jar/META-INF/maven/org.springframework.cloud/spring-cloud-consul-config/pom.xml

1.153 aws-java-sdk-s3 1.11.415

1.153.1 Available under license:
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2014-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
* Portions copyright 2006-2009 James Murty. Please see LICENSE.txt
* for applicable license terms and NOTICE.txt for applicable notices.
* Licensed under the Apache License, Version 2.0 (the "License").
* You may not use this file except in compliance with the License.
* A copy of the License is located at
* http://aws.amazon.com/apache2.0
*
* or in the "license" file accompanying this file. This file is distributed
* on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express or implied. See the License for the specific language governing
* permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/services/s3/internal/crypto/SecuredCEK.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/services/s3/internal/crypto/KMSSecuredCEK.java

No license file was found, but licenses were detected in source scan.

/
* Copyright 2010-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
* Licensed under the Apache License, Version 2.0 (the "License").
* You may not use this file except in compliance with the License.
* A copy of the License is located at
* http://aws.amazon.com/apache2.0
*
* or in the "license" file accompanying this file. This file is distributed
* on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express or implied. See the License for the specific language governing
* permissions and limitations under the License.
*/
/**
 * Gets the optional <code>versionIdMarker</code> parameter indicating where in the
 * sorted list of all versions in the specified bucket to begin returning
 * results. Results are always ordered first lexicographically (i.e.
 * alphabetically) and then from most recent version to least recent
 * version.
 * <p>
 * A key marker must be specified when specifying a <code>versionIdMarker</code>.
 * Results begin immediately after the version with the specified key and
 * version ID.
 * </p>
 *
 * @return The optional <code>versionIdMarker</code> parameter indicating where in the
 * sorted list of all versions in the specified bucket to begin
 * returning results.
 *
 * @see ListVersionsRequest#setVersionIdMarker(String)
*/

Found in path(s):
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
  jar/com/amazonaws/services/s3/model/ListVersionsRequest.java

No license file was found, but licenses were detected in source scan.

/*/ 

* Copyright 2010-2018 Amazon.com, Inc. or its affiliates. All Rights
  Reserved.

* Licensed under the Apache License, Version 2.0 (the "License").
* You may not use this file except in compliance with the License.
* A copy of the License is located at
* 
* http://aws.amazon.com/apache2.0
*
* or in the "license" file accompanying this file. This file is distributed
* on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express or implied. See the License for the specific language governing
* permissions and limitations under the License.

*/

Found in path(s):
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
  jar/com/amazonaws/services/s3/internal/auth/S3SignerProvider.java

No license file was found, but licenses were detected in source scan.

/*/ 

* Copyright (c) 2016. Amazon.com, Inc. or its affiliates. All Rights Reserved.
* 
* Licensed under the Apache License, Version 2.0 (the "License").
* You may not use this file except in compliance with the License.
* A copy of the License is located at
* 
* http://aws.amazon.com/apache2.0
*
* or in the "license" file accompanying this file. This file is distributed
* on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express or implied. See the License for the specific language governing
* permissions and limitations under the License.

*/

Found in path(s):
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
  jar/com/amazonaws/services/s3/internal/SkipMd5CheckStrategy.java
  /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 * Licensed under the Apache License, Version 2.0 (the "License").
 * You may not use this file except in compliance with the License.
 * A copy of the License is located at
 * http://aws.amazon.com/apache2.0
 * or in the "license" file accompanying this file. This file is distributed
 * on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing
 * permissions and limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
  jar/com/amazonaws/services/s3/model/BucketAccelerateConfiguration.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
  jar/com/amazonaws/services/s3/model/transform/ProgressStaxUnmarshaller.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
  jar/com/amazonaws/services/s3/model/GetBucketNotificationConfigurationRequest.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
  jar/com/amazonaws/services/s3/model/transform/S3KeyFilterStaxUnmarshaller.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
  jar/com/amazonaws/services/s3/model/ReplicationRuleStatus.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2017-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
* Licensed under the Apache License, Version 2.0 (the "License").
* You may not use this file except in compliance with the License.
* A copy of the License is located at
* http://aws.amazon.com/apache2.0
*
* or in the "license" file accompanying this file. This file is distributed
* on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express or implied. See the License for the specific language governing
* permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/services/s3/internal/eventstreaming/Header.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/services/s3/internal/eventstreaming/Checksums.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/services/s3/internal/eventstreaming/Prelude.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/services/s3/internal/eventstreaming/HeaderValue.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/services/s3/internal/eventstreaming/Message.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/services/s3/internal/eventstreaming/HeaderType.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/services/s3/internal/eventstreaming/Utils.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2010-20184 Amazon.com, Inc. or its affiliates. All Rights Reserved.
* Licensed under the Apache License, Version 2.0 (the "License").
* You may not use this file except in compliance with the License.
* A copy of the License is located at
* http://aws.amazon.com/apache2.0
* or in the "license" file accompanying this file. This file is distributed
* on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express or implied. See the License for the specific language governing
* permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
/*
 * Copyright 2010-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 * Licensed under the Apache License, Version 2.0 (the "License").
 * You may not use this file except in compliance with the License.
 * A copy of the License is located at
 * http://aws.amazon.com/apache2.0
 *
 * or in the "license" file accompanying this file. This file is distributed
 * on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing
 * permissions and limitations under the License.
 */

// The Amazon S3 DevPay token header is a special exception and can be safely moved

Found in path(s):
*/
/*
 * Copyright 2011-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 * Licensed under the Apache License, Version 2.0 (the "License").
 * You may not use this file except in compliance with the License.
 * A copy of the License is located at
 * http://aws.amazon.com/apache2.0
 *
 * or in the "license" file accompanying this file. This file is distributed
 * on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing
 * permissions and limitations under the License.
 */

Found in path(s):
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2010-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 *
 * Portions copyright 2006-2009 James Murty. Please see LICENSE.txt
 * for applicable license terms and NOTICE.txt for applicable notices.
 *
 * Licensed under the Apache License, Version 2.0 (the "License").
 * You may not use this file except in compliance with the License.
 * A copy of the License is located at
 *
 * http://aws.amazon.com/apache2.0
 *
 * or in the "license" file accompanying this file. This file is distributed
 * on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing
 * permissions and limitations under the License.
 */

Found in path(s):

* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-3-1-11-415-sources-1.jar/com/amazonaws/services/s3/AmazonS3ClientParams.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-3-1-11-415-sources-1.jar/com/amazonaws/services/s3/model/RoutingRule.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-3-1-11-415-sources-1.jar/com/amazonaws/services/s3/model/ListObjectsV2Result.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-3-1-11-415-sources-1.jar/com/amazonaws/services/s3/internal/S3AbortableInputStream.java

* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-3-1-11-415-sources-1.jar/com/amazonaws/services/s3/model/S3AbortableInputStream.java

* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-3-1-11-415-sources-1.jar/com/amazonaws/services/s3/internal/S3AbortableInputStream.java

* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-3-1-11-415-sources-1.jar/com/amazonaws/services/s3/internal/S3AbortableInputStream.java

* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-3-1-11-415-sources-1.jar/com/amazonaws/services/s3/internal/S3AbortableInputStream.java

* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-3-1-11-415-sources-1.jar/com/amazonaws/services/s3/internal/S3AbortableInputStream.java

* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-3-1-11-415-sources-1.jar/com/amazonaws/services/s3/internal/S3AbortableInputStream.java

* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-3-1-11-415-sources-1.jar/com/amazonaws/services/s3/internal/S3AbortableInputStream.java

* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-3-1-11-415-sources-1.jar/com/amazonaws/services/s3/internal/S3AbortableInputStream.java

* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-3-1-11-415-sources-1.jar/com/amazonaws/services/s3/internal/S3AbortableInputStream.java

* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-3-1-11-415-sources-1.jar/com/amazonaws/services/s3/internal/S3AbortableInputStream.java

* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-3-1-11-415-sources-1.jar/com/amazonaws/services/s3/internal/S3AbortableInputStream.java

* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-3-1-11-415-sources-1.jar/com/amazonaws/services/s3/internal/S3AbortableInputStream.java

* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-3-1-11-415-sources-1.jar/com/amazonaws/services/s3/internal/S3AbortableInputStream.java

* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-3-1-11-415-sources-1.jar/com/amazonaws/services/s3/internal/S3AbortableInputStream.java

* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-3-1-11-415-sources-1.jar/com/amazonaws/services/s3/internal/S3AbortableInputStream.java

* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-3-1-11-415-sources-1.jar/
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2012-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 *
 * Licensed under the Apache License, Version 2.0 (the "License").
 * You may not use this file except in compliance with the License.
 * A copy of the License is located at
 *
 *  http://aws.amazon.com/apache2.0
 *
 * or in the "license" file accompanying this file. This file is distributed
 * on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing
 * permissions and limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1.jar/com/amazonaws/services/s3/iterable/S3Versions.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1.jar/com/amazonaws/services/s3/transfer/Download.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1.jar/com/amazonaws/services/s3/model/TagSet.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1.jar/com/amazonaws/services/s3/internal/RepeatableInputStream.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1.jar/com/amazonaws/services/s3/model/APIException.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1.jar/com/amazonaws/services/s3/model/BucketNotificationConfiguration.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1.jar/com/amazonaws/services/s3/model/Bucket.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1.jar/com/amazonaws/services/s3/model/BucketPolicy.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1.jar/com/amazonaws/services/s3/model/CanonicalGrantee.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1.jar/com/amazonaws/services/s3/model/EmailAddressGrantee.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1.jar/com/amazonaws/services/s3/model/Grantee.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1.jar/com/amazonaws/services/s3/internal/ServiceUtils.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1.jar/com/amazonaws/services/s3/model/TagSet.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1.jar/com/amazonaws/services/s3/model/BucketPolicy.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1.jar/com/amazonaws/services/s3/model/TagSet.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1.jar/com/amazonaws/services/s3/model/BucketPolicy.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1.jar/com/amazonaws/services/s3/model/TagSet.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1.jar/com/amazonaws/services/s3/model/BucketPolicy.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1.jar/com/amazonaws/services/s3/model/TagSet.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1.jar/com/amazonaws/services/s3/model/BucketPolicy.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1.jar/com/amazonaws/services/s3/model/TagSet.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1.jar/com/amazonaws/services/s3/model/BucketPolicy.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1.jar/com/amazonaws/services/s3/model/TagSet.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2012-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 *
 * Licensed under the Apache License, Version 2.0 (the "License"). You may not use this file except in compliance with
 * the License. A copy of the License is located at
 *
 * http://aws.amazon.com/apache2.0
 *
 * or in the "license" file accompanying this file. This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR
 * CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions
 * and limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/services/s3/model/inventory/InventoryEncryption.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/services/s3/model/DeleteMarkerReplication.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/services/s3/model/SseKmsEncryptedObjectsStatus.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2011-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 *
 * Licensed under the Apache License, Version 2.0 (the "License);
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at:
 *
 *    http://aws.amazon.com/apache2.0
 *
 * This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES
 * OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
*/

* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-3-1-11-415-sources-1-jar/com/amazonaws/services/s3/model/DeleteBucketEncryptionResult.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-3-1-11-415-sources-1-jar/com/amazonaws/services/s3/model/ServerSideEncryptionKMS.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/services/s3/model/SseKmsEncryptedObjects.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/services/s3/model/SetBucketInventoryConfigurationRequest.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-3-1-11-415-sources-1-jar/com/amazonaws/services/s3/model/metrics/MetricsPredicateVisitor.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-3-1-11-415-sources-1-jar/com/amazonaws/services/s3/model/ListBucketAnalyticsConfigurationsRequest.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/services/s3/model/metrics/MetricsConfiguration.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-3-1-11-415-sources-1-jar/com/amazonaws/services/s3/model/lifecycle/LifecycleAndOperator.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-3-1-11-415-sources-1-jar/com/amazonaws/services/s3/model/GetBucketMetricsConfigurationResult.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-3-1-11-415-sources-1-jar/com/amazonaws/services/s3/model/analytics/AnalyticsAndOperator.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-3-1-11-415-sources-1-jar/com/amazonaws/services/s3/model/replication/ReplicationNaryOperator.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-3-1-11-415-sources-1-jar/com/amazonaws/services/s3/model/SetBucketMetricsConfigurationRequest.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-3-1-11-415-sources-1-jar/com/amazonaws/services/s3/model/SetBucketInventoryConfigurationRequest.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2012-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 *
 * Licensed under the Apache License, Version 2.0 (the "License").
 * You may not use this file except in compliance with the License.
 * A copy of the License is located at
 * http://aws.amazon.com/apache2.0
Open Source Used In security_automation identity-guest-cs 860

* or in the "license" file accompanying this file. This file is distributed
* on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express or implied. See the License for the specific language governing
* permissions and limitations under the License.
*/
/**
* Interface for providing encryption materials.
* Implementations are free to use any strategy for providing encryption
* materials, such as simply providing static material that doesn't change,
* or more complicated implementations, such as integrating with existing
* key management systems.
*/

Found in path(s):
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
jar/com/amazonaws/services/s3/model/EncryptionMaterialsProvider.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2013-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
* Licensed under the Apache License, Version 2.0 (the "License").
* You may not use this file except in compliance with the License.
* A copy of the License is located at
* http://aws.amazon.com/apache2.0
*
* or in the "license" file accompanying this file. This file is distributed
* on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express or implied. See the License for the specific language governing
* permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
jar/com/amazonaws/services/s3/transfer/PauseResult.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2013-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
* Licensed under the Apache License, Version 2.0 (the "License").
* You may not use this file except in compliance with the License.
* A copy of the License is located at
* http://aws.amazon.com/apache2.0
*
* or in the "license" file accompanying this file. This file is distributed
* on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express or implied. See the License for the specific language governing
* permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
jar/com/amazonaws/services/s3/internal/crypto/ContentCryptoMaterial.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
jar/com/amazonaws/services/s3/internal/crypto/MultipartUploadCryptoContext.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
jar/com/amazonaws/services/s3/internal/crypto/CryptoRuntime.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
jar/com/amazonaws/services/s3/internal/crypto/S3KeyWrapScheme.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
jar/com/amazonaws/services/s3/internal/crypto/S3CryptoModuleBase.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
jar/com/amazonaws/services/s3/internal/crypto/AesGcm.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
jar/com/amazonaws/services/s3/internal/crypto/S3CryptoModuleAE.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
jar/com/amazonaws/services/s3/transfer/exception/PauseException.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
jar/com/amazonaws/services/s3/transfer/PersistableTransfer.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
jar/com/amazonaws/services/s3/transfer/internal/S3ProgressPublisher.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
jar/com/amazonaws/services/s3/internal/crypto/MultipartUploadCbcContext.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
jar/com/amazonaws/services/s3/internal/crypto/S3ProgressListener.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
jar/com/amazonaws/services/s3/AmazonS3Encryption.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
jar/com/amazonaws/services/s3/internal/crypto/CipherLiteInputStream.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
jar/com/amazonaws/services/s3/internal/crypto/S3CryptoModuleAEStrict.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
jar/com/amazonaws/services/s3/transfer/PersistentUpload.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
jar/com/amazonaws/services/s3/transfer/internal/S3ProgressListenerChain.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
jar/com/amazonaws/services/s3/internal/crypto/AesCbc.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
jar/com/amazonaws/services/s3/AmazonS3EncryptionClient.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
jar/com/amazonaws/services/s3/internal/crypto/ContentCryptoScheme.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
* for applicable license terms and NOTICE.txt for applicable notices.
*/

Found in path(s):
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/services/s3/model/S3Object.java
No license file was found, but licenses were detected in source scan.

*/

* Copyright 2014-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License").
* You may not use this file except in compliance with the License.
* A copy of the License is located at
* http://aws.amazon.com/apache2.0
* or in the "license" file accompanying this file. This file is distributed
* on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express or implied. See the License for the specific language governing
* permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/services/s3/model/S3ObjectId.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/auth/DecodedStreamBuffer.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/services/s3/model/PutInstructionFileRequest.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/services/s3/model/SimpleMaterialProvider.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/services/s3/model/InstructionFileId.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/services/s3/model/NotificationConfiguration.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/services/s3/model/SSECustomerKeyProvider.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/services/s3/model/KeyWrapException.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/services/s3/model/S3DataSource.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/services/s3/internal/FileDeletionEvent.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1.jar/com/amazonaws/services/s3/internal/AmazonS3ExceptionBuilder.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1.jar/com/amazonaws/services/s3/internal/MultiFileOutputStream.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1.jar/com/amazonaws/services/s3/internal/S3Direct.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1.jar/com/amazonaws/services/s3/OnFileDelete.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1.jar/com/amazonaws/services/s3/model/EncryptedPutObjectRequest.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1.jar/com/amazonaws/services/s3/model/S3ObjectIddBuilder.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1.jar/com/amazonaws/services/s3/transfer/internal/S3SyncProgressListener.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1.jar/com/amazonaws/services/s3/transfer/MultipleFileTransferStateChangeListener.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1.jar/com/amazonaws/services/s3/model/S3DirectSpi.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1.jar/com/amazonaws/services/s3/model/EncryptedInitiateMultipartUploadRequest.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1.jar/com/amazonaws/services/s3/model/MaterialsDescriptionProvider.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1.jar/com/amazonaws/services/s3/model/PartCreationEvent.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1.jar/com/amazonaws/services/s3/model/MultipleFileTransferProgressUpdatingListener.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1.jar/com/amazonaws/services/s3/model/PutObjectRequest.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1.jar/com/amazonaws/services/s3/AmazonS3URI.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1.jar/com/amazonaws/services/s3/model/CloudFunctionConfiguration.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1.jar/com/amazonaws/services/s3/model/SSECustomerKey.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1.jar/com/amazonaws/services/s3/model/ExtraMaterialsDescription.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015-2018 Amazon Technologies, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at:
 *
 * http://aws.amazon.com/apache2.0
 *
 * This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES
 * OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/services/s3/model/ListNextBatchOfVersionsRequest.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2017. Amazon.com, Inc. or its affiliates. All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License").
* You may not use this file except in compliance with the License.
* A copy of the License is located at
*
* http://aws.amazon.com/apache2.0
*
* or in the "license" file accompanying this file. This file is distributed
* on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express or implied. See the License for the specific language governing
* permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/services/s3/AmazonS3EncryptionClientParamsWrapper.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/services/s3/AmazonS3EncryptionClientParams.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/services/s3/transfer/KeyFilter.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/services/s3/AmazonS3EncryptionClientBuilder.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2011-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License"). You may not
* use this file except in compliance with the License. A copy of the License is
* located at
*
* http://aws.amazon.com/apache2.0
*
* or in the "license" file accompanying this file. This file is distributed on
* an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express or implied. See the License for the specific language governing
* permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/services/s3/waiters/HeadObjectFunction.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2012-2018 Amazon Technologies, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at:
 *
 *   http://aws.amazon.com/apache2.0
 *
 * This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES
 * OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/services/s3/waiters/HeadBucketFunction.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/services/s3/transfer/internal/DownloadMonitor.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/services/s3/transfer/MultipleFileUpload.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/services/s3/transfer/internal/TransferStateChangeListener.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/services/s3/transfer/internal/MultipleFileUploadImpl.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/services/s3/transfer/internal/MultipleFileTransferMonitor.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/services/s3/transfer/internal/MultipleFileTransfer.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/services/s3/model/S3ObjectInputStream.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/services/s3/transfer/internal/MultipleFileDownloadImpl.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014-2018 Amazon Technologies, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at:
 */
* http://aws.amazon.com/apache2.0
*
* This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES
* OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
  jar/com/amazonaws/services/s3/event/S3EventNotification.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2016-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License").
* You may not use this file except in compliance with the License.
* A copy of the License is located at
*
* http://aws.amazon.com/apache2.0
*
* or in the "license" file accompanying this file. This file is distributed
* on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express or implied. See the License for the specific language governing
* permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
  jar/com/amazonaws/services/s3/model/GetObjectTaggingResult.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
  jar/com/amazonaws/services/s3/model/GetObjectTaggingRequest.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
  jar/com/amazonaws/services/s3/model/GetObjectAclRequest.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
  jar/com/amazonaws/services/s3/model/DeleteObjectTaggingHeaderHandler.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
  jar/com/amazonaws/services/s3/model/DeleteObjectTaggingResult.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
  jar/com/amazonaws/services/s3/model/SetObjectTaggingRequest.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
  jar/com/amazonaws/services/s3/model/Tag.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
  jar/com/amazonaws/services/s3/model/transform/ObjectTaggingXmlFactory.java
* /opt/ cola/ permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/services/s3/internal/S3RequesterChargedHeaderHandler.java
* /opt/ cola/ permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/services/s3/model/S3AccelerateUnsupported.java
* /opt/ cola/ permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/services/s3/model/SetObjectTaggingResult.java
* /opt/ cola/ permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/services/s3/model/SetBucketLifecycleConfigurationRequest.java
* /opt/ cola/ permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/services/s3/model/S3ObjectListing.java
* /opt/ cola/ permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/services/s3/model/BucketVersioningConfiguration.java
* /opt/ cola/ permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/services/s3/model/SetBucketVersioningConfigurationRequest.java
* /opt/ cola/ permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/services/s3/model/ObjectTagging.java
* /opt/ cola/ permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/services/s3/model/S3ObjectSummary.java

No license file was found, but licenses were detected in source scan.

/ *
* Copyright 2010-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
* *
* Licensed under the Apache License, Version 2.0 (the "License").
* You may not use this file except in compliance with the License.
* A copy of the License is located at
* *
* http://aws.amazon.com/apache2.0
* *
* or in the "license" file accompanying this file. This file is distributed
* on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express or implied. See the License for the specific language governing
* permissions and limitations under the License.
*/

Found in path(s):
* /opt/ cola/ permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/services/s3/model/SetBucketLifecycleConfigurationRequest.java
* /opt/ cola/ permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/services/s3/model/ObjectListing.java
* /opt/ cola/ permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/services/s3/model/SetBucketVersioningConfigurationRequest.java
* /opt/ cola/ permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/services/s3/model/BucketVersioningConfiguration.java
* /opt/ cola/ permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/services/s3/model/SetBucketTaggingConfigurationRequest.java
* /opt/ cola/ permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/services/s3/model/MultiFactorAuthentication.java
* /opt/ cola/ permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/services/s3/model/S3ObjectSummary.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2011-2018 Amazon Technologies, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at:
 *
 *    http://aws.amazon.com/apache2.0
 *
 * This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES
 * OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
  jar/com/amazonaws/services/s3/model/GenericBucketRequest.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
  jar/com/amazonaws/services/s3/model/ResponseHeaderHandlerChain.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
  jar/com/amazonaws/services/s3/model/MD5DigestCalculatingInputStream.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
  jar/com/amazonaws/services/s3/model/package-info.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
  jar/com/amazonaws/services/s3/model/ServerSideEncryptionResult.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
  jar/com/amazonaws/services/s3/model/DeleteObjectsResponse.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
  jar/com/amazonaws/services/s3/transfer/internal/CopyMonitor.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
  jar/com/amazonaws/services/s3/transfer/internal/ServerSideEncryptionResult.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
  jar/com/amazonaws/services/s3/package-info.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1.jar/com/amazonaws/services/s3/model/CopyPartRequest.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1.jar/com/amazonaws/services/s3/model/HeaderHandler.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1.jar/com/amazonaws/services/s3/model/CORSRule.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1.jar/com/amazonaws/services/s3/model/BucketLifecycleConfiguration.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1.jar/com/amazonaws/services/s3/model/MultiObjectDeleteException.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1.jar/com/amazonaws/services/s3/model/CopyPartResult.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1.jar/com/amazonaws/services/s3/model/ObjectExpirationResult.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2013-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 * Portions copyright 2006-2009 James Murty. Please see LICENSE.txt
 * for applicable license terms and NOTICE.txt for applicable notices.
 * Licensed under the Apache License, Version 2.0 (the "License").
 * You may not use this file except in compliance with the License.
 * A copy of the License is located at
 * http://aws.amazon.com/apache2.0
 * or in the "license" file accompanying this file. This file is distributed
 * on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing
 * permissions and limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1.jar/com/amazonaws/services/s3/internal/AbstractRepeatableCipherInputStream.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2016-2018 Amazon Technologies, Inc.
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at:
* http://aws.amazon.com/apache2.0
* This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/services/s3/model/AbortIncompleteMultipartUpload.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/services/s3/internal/S3RequesterChargedResult.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2010-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
* Licensed under the Apache License, Version 2.0 (the "License"). You may not use this file except in compliance with the License. A copy of the License is located at
* http://aws.amazon.com/apache2.0
* or in the "license" file accompanying this file. This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/services/s3/internal/S3RequestEndpointResolver.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/services/s3/model/ListMultipartUploadsRequest.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/services/s3/model/SetRequestPaymentConfigurationRequest.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/services/s3/model/DeleteObjectRequest.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-jar/com/amazonaws/services/s3/model/S3RestoreOutputPathHeaderHandler.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
Open Source Used In security_automation identity-guest-cs 874
jar/com/amazonaws/services/s3/internal/S3ObjectResponseHandler.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
jar/com/amazonaws/services/s3/model/GeneratePresignedUrlRequest.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
jar/com/amazonaws/services/s3/internal/SSEResultBase.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
jar/com/amazonaws/services/s3/model/AmazonS3Exception.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
jar/com/amazonaws/services/s3/transfer/UploadContext.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
jar/com/amazonaws/services/s3/model/MultipartUpload.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
jar/com/amazonaws/services/s3/model/transform/RequestPaymentConfigurationXmlFactory.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
jar/com/amazonaws/services/s3/model/OutputSerialization.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
jar/com/amazonaws/services/s3/model/RestoreObjectRequest.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
jar/com/amazonaws/services/s3/model/CompleteMultipartUploadRequest.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
jar/com/amazonaws/services/s3/transfer/TransferManager.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
jar/com/amazonaws/services/s3/internal/S3MetadataResponseHandler.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
jar/com/amazonaws/services/s3/model/PartSummary.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
jar/com/amazonaws/services/s3/internal/crypto/JceEncryptionConstants.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
jar/com/amazonaws/services/s3/model/MetadataDirective.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
jar/com/amazonaws/services/s3/transfer/internal/UploadCallable.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
jar/com/amazonaws/services/s3/model/MetadataEntry.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
jar/com/amazonaws/services/s3/model/Region.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
jar/com/amazonaws/services/s3/transfer/model/CopyResult.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
jar/com/amazonaws/services/s3/model/QuoteFields.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
jar/com/amazonaws/services/s3/model/ProgressListener.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
jar/com/amazonaws/services/s3/transfer/ObjectTaggingProvider.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
jar/com/amazonaws/services/s3/transfer/package-info.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
jar/com/amazonaws/services/s3/transfer/transfer/package-info.java
jar/com/amazonaws/services/s3/transfer/internal/CopyPartRequestFactory.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
jar/com/amazonaws/services/s3/transfer/internal/TransferProgressUpdatingListener.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
jar/com/amazonaws/services/s3/model/CannedAccessControlList.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
jar/com/amazonaws/services/s3/internal/BucketNameUtils.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
jar/com/amazonaws/services/s3/model/PutObjectResult.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
jar/com/amazonaws/services/s3/model/CSVInput.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
jar/com/amazonaws/services/s3/metrics/S3ServiceMetric.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
jar/com/amazonaws/auth/policy/resources/S3BucketResource.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
jar/com/amazonaws/services/s3/model/AbortMultipartUploadRequest.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
jar/com/amazonaws/services/s3/internal/DualstackEndpointBuilder.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
jar/com/amazonaws/services/s3/model/GetObjectPaymentConfigurationRequest.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
jar/com/amazonaws/services/s3/model/AbortMultipartUploadRequest.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
jar/com/amazonaws/services/s3/model/LifecyclePredicateVisitorImpl.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
jar/com/amazonaws/services/s3/model/transform/AclXmlFactory.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
jar/com/amazonaws/services/s3/model/Encryption.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
jar/com/amazonaws/services/s3/model/transform/InternalCopyPartCallable.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
jar/com/amazonaws/services/s3/model/transform/LifecyclePredicateVisitorImpl.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-
jar/com/amazonaws/services/s3/model/PartETag.java
* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-1-11-415-sources-1-

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2016-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 * Portions copyright 2006-2009 James Murty. Please see LICENSE.txt
 * for applicable license terms and NOTICE.txt for applicable notices.
 * Licensed under the Apache License, Version 2.0 (the "License").
 * You may not use this file except in compliance with the License.
 * A copy of the License is located at
 * http://aws.amazon.com/apache2.0
 * or in the "license" file accompanying this file. This file is distributed
 * on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing
 * permissions and limitations under the License.
 */

Found in path(s):

/*
 * Copyright 2011-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 * Licensed under the Apache License, Version 2.0 (the "License"). You may not
 * use this file except in compliance with the License. A copy of the License is
 * located at
 * http://aws.amazon.com/apache2.0
 * or in the "license" file accompanying this file. This file is distributed on
 * an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing
 * permissions and limitations under the License.
 */

Found in path(s):
/*
 * Copyright 2010-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 *
 * Licensed under the Apache License, Version 2.0 (the "License").
 * You may not use this file except in compliance with the License.
 * A copy of the License is located at
 *
 * http://aws.amazon.com/apache2.0
 *
 * or in the "license" file accompanying this file. This file is distributed
 * on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing
 * permissions and limitations under the License.
 */

/*
 Synchronous client for accessing Amazon S3.
 *
 * <h1>Appendix: Amazon S3 client-side encryption meta information</h1>
 *
 * <h2>Introduction</h2>
 *
 * This appendix summarizes the current crypto related meta information
 * associated with an S3 object encrypted using SDK client-side encryption. In
 * SDK's terminology, these meta information are classified as "user metadata".
 * In general, the meta information is just a map of key/value pairs. Physically
 * they are persisted either as metadata of the encrypted S3 object, or as a
 * separate S3 object called an "instruction file" (which is just a JSON file in
 * plaintext.) For all references to the Cipher Algorithm Names, Modes, and Padding,
 * such as <code>"AES/CBC/PKCS5Padding"</code>, please refer to Oracle's <a
 * href="http://docs.oracle.com/javase/7/docs/technotes/guides/security/StandardNames.html"
 * >Java® Cryptography Architecture Standard Algorithm Name Documentation</a>
 * for more details.
 *
 * <h2>S3 metadata vs instruction file</h2>
 *
 * Note when a key/value pair is stored as "user metadata" as part of the S3
 * metadata, the SDK always adds an additional prefix of <code>x-amz-meta-</code> to
 * the key name. In contrast, the SDK doesn't add any prefix when the key names
 * are stored in an instruction file. (The reason has to do with how the http
 * headers are handled.)
 *
 * For example, <code>&lt;b&gt;x-amz-key&lt;/b&gt;</code> would get stored as <code>&lt;b&gt;x-amz-meta-x-amz-key&lt;/b&gt;</code>
 * in S3 metadata. In contrast, <code>&lt;b&gt;x-amz-key&lt;/b&gt;</code> would simply be stored as is
 * (ie without the prefix) in an instruction file.
 */
* All the keys described below are considered "user metadata" by the SDK.

* <h2>S3 v1 crypto meta information (supported by all SDK's)</h2>

* For non-KMS client side encryption, this v1 crypto meta information is used
* to support the "Encryption Only" crypto mode (which involves the use of
* <code>AES/CBC/PKCS5Padding</code> for content encryption, and
* <code>AES/ECB</code> for encrypting the one-time randomly generated data
* key). For KMS client-side encryption, the v2 crypto meta information is used.

* <p>
* <table border=1>
* <tr aligned="left">
* <th>key</th>
* <th>description</th>
* </tr>
* <tr>
* <td width="200"><b>x-amz-key</b></td>
* <td>Content encrypting key (cek) in encrypted form, base64 encoded. The cek
* is randomly generated per S3 object, and is always an AES 256-bit key. The
* corresponding cipher is always "AES/CBC/PKCS5Padding".</td>
* </tr>
* <tr>
* <td><b>x-amz-iv</b></td>
* <td>Randomly generated IV (per S3 object), base64 encoded</td>
* </tr>
* <tr>
* <td><b>x-amz-matdesc </b></td>
* <td>Customer provided material description in JSON (UTF8) format. Used to
* identify the client-side master key (ie used to encrypt/wrap the generated
* content encrypting key).</td>
* </tr>
* <tr>
* <td><b>x-amz-unencrypted-content-length</b></td>
* <td>Unencrypted content length (optional but should be specified whenever
* possible).</td>
* </tr>
* </table>

* <h2>S3 v2 crypto meta information (currently supported by the Java SDK)</h2>

* This v2 crypto meta information is used to support both authenticated
* encryption (which involves the use of <code>AES/GCM/NoPadding</code> for
* content encryption, and key wrapping for the one-time randomly generated data
* key), and KMS client-side encryption (which can either be encryption-only or
* authenticated encryption.)

* <p>
* <table border=1>
<table>
<thead>
<tr>
<th>key</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>x-amz-key-v2</td>
<td>CEK in key wrapped form. This is necessary so that the S3 encryption client that doesn't recognize the v2 format will not mistakenly decrypt S3 object encrypted in v2 format.</td>
</tr>
<tr>
<td>x-amz-iv</td>
<td>Randomly generated IV (per S3 object), base64 encoded. (Same as v1.)</td>
</tr>
<tr>
<td>x-amz-matdesc</td>
<td>Customer provided material description in JSON format. (Same as v1). Used to identify the client-side master key. For KMS client side encryption, the KMS Customer Master Key ID is stored as part of the material description, &lt;b&gt;x-amz-matdesc&lt;/b&gt;, under the key-name &lt;code&gt;&quot;kms_cmk_id&quot;&lt;/code&gt;</td>
</tr>
<tr>
<td>x-amz-unencrypted-content-length</td>
<td>Unencrypted content length. (optional but should be specified whenever possible. Same as v1.)</td>
</tr>
<tr>
<td>x-amz-wrap-alg</td>
<td>Key wrapping algorithm used. Supported values: No standard key wrapping is used if this meta information is absent. Always set to &lt;b&gt;&quot;kms&quot;&lt;/b&gt; if KMS is used for client-side encryption.</td>
</tr>
<tr>
<td>x-amz-cek-alg</td>
<td>Content encryption algorithm used. Supported values: Default to &lt;code&gt;&quot;AES/CBC/PKCS5Padding&quot;&lt;/code&gt; if this key is absent.</td>
</tr>
</tbody>
</table>
* <ul>
*     * </ul>
* <table><tr><td><b>x-amz-tag-len</b></td><td>Tag length (in bits) when AEAD is in use.<br/>
*     * </td></tr><tr><td><b>x-amz-wrap-alg</b></td><td>Only applicable if AEAD is in use. This meta information is absent otherwise, or if KMS is in use.<br/>
*     * </td></tr><tr><td><b>x-amz-matdesc</b></td><td>Supported value: <code>"128"</code><br/>
*     * </td></tr><tr><td><b>x-amz-meta-x-amz-key-v2</b></td><td><pre>CiAJ2GqwpJRnt4izujwoailC2YdNRNMSlGlG5rw3pEI+J8hKnAQEBAwB4CdhsqKSUZ7eIs7o8KGpQtmHTUTUjBLJRua86RCPft1AAAB+MHwGCSqGSlb3DQEHBqBvMG0CAQAwAAYJKoZlhrvCNACQCwCMB4GCWCGSAFlAwQBLjARBAyD0rzN3KHViwixDVcCARCAOwhZ6xA8rob3Z0mNx3uKFeiHKVyiIC/PK1chALvG1oVLnf86mcZJNpTXtwMkKkDocVj3Z1zGjZTuF5FSOH
x-amz-meta-x-amz-key-v2:
* x-amz-meta-x-amz-key-v2:
* CiAJ2GqwpJRnt4izujwoailC2YdNRNMSlGlG5rw3pEI+J8hKnAQEBAwB4CdhsqKSUZ7eIs7o8KGpQtmHTUTUj
</pre>

<h3>KMS Integration for client-side encryption</h3>

* All client-side KMS protected S3 objects are stored in v2 crypto meta information format. However, the key wrapping algorithm, <b>x-amz-wrap-alg</b> is always set to <b>"kms"</b>.<br/>
* The KMS Customer Master Key ID is currently stored as part of the material description, <b>x-amz-matdesc</b>, under the key-name <b>x-amz-cmk_id</b>.<br/>

<h4>Sample S3 crypto meta information for KMS client-side encryption</h4>

**CryptoMode: Encryption Only**

```plaintext
x-amz-meta-x-amz-key-v2:
* x-amz-meta-x-amz-key-v2:
CiAJ2GqwpJRnt4izujwoailC2YdNRNMSlGlG5rw3pEI+J8hKnAQEBAwB4CdhsqKSUZ7eIs7o8KGpQtmHTUTUjBLJRua86RCPft1AAAB+MHwGCSqGSlb3DQEHBqBvMG0CAQAwAAYJKoZlhrvCNACQCwCMB4GCWCGSAFlAwQBLjARBAyD0rzN3KHViwixDVcCARCAOwhZ6xA8rob3Z0mNx3uKFeiHKVyiIC/PK1chALvG1oVLnf86mcZJNpTXtwMkKkDocVj3Z1zGjZTuF5FSOH
```

**CryptoMode: Authenticated Encryption**

```plaintext
x-amz-meta-x-amz-key-v2:
* x-amz-meta-x-amz-key-v2:
CiAJ2GqwpJRnt4izujwoailC2YdNRNMSlGlG5rw3pEI+J8hKnAQEBAwB4CdhsqKSUZ7eIs7o8KGpQtmHTUTUj
```
The following key is always included as part of the S3 metadata of the instruction file (which is just an S3 object). Since the SDK treats this key as "user metadata", the key name is always prefixed with <b>x-amz-meta-</b> when physically stored.

<table>
<thead>
<tr>
<th>key</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>x-amz-crypto-instr-file</td>
<td>the presence of this key is used to indicate this S3 object is an instruction file (of an associated client-side encrypted S3 object).</td>
</tr>
</tbody>
</table>

Found in path(s):

* /opt/cola/permits/1136114796_1613679646.47/0/aws-java-sdk-s3-3.1.11-415-sources-1.jar/com/amazonaws/services/s3/package-info.java

1.154 bean-validation-api 2.0.1.Final

1.154.1 Available under license:

Bean Validation API

License: Apache License, Version 2.0

See the license.txt file in the root directory or <http://www.apache.org/licenses/LICENSE-2.0>.

1.155 msgpack-core 0.8.20
1.155.1 Available under license:

No license file was found, but licenses were detected in source scan.

// Licensed under the Apache License, Version 2.0 (the "License");
// you may not use this file except in compliance with the License.
// You may obtain a copy of the License at
//    http://www.apache.org/licenses/LICENSE-2.0
// distributed under the License is distributed on an "AS IS" BASIS,

Found in path(s):
* /opt/cola/permits/1135840621_1613612884.82/0/msgpack-core-0-8-20-sources-1-jar/org/msgpack/core/MessagePack.java
* /opt/cola/permits/1135840621_1613612884.82/0/msgpack-core-0-8-20-sources-1-jar/org/msgpack/value/impl/AbstractImmutableValue.java
* /opt/cola/permits/1135840621_1613612884.82/0/msgpack-core-0-8-20-sources-1-jar/org/msgpack/core/buffer/ChannelBufferOutput.java
* /opt/cola/permits/1135840621_1613612884.82/0/msgpack-core-0-8-20-sources-1-jar/org/msgpack/value/ImmutableValue.java
* /opt/cola/permits/1135840621_1613612884.82/0/msgpack-core-0-8-20-sources-1-jar/org/msgpack/value/impl/ImmutableValueImpl.java
* /opt/cola/permits/1135840621_1613612884.82/0/msgpack-core-0-8-20-sources-1-jar/org/msgpack/core/buffer/DirectBufferAccess.java
* /opt/cola/permits/1135840621_1613612884.82/0/msgpack-core-0-8-20-sources-1-jar/org/msgpack/value/ExtensionValue.java
* /opt/cola/permits/1135840621_1613612884.82/0/msgpack-core-0-8-20-sources-1-jar/org/msgpack/value/ValueType.java
Open Source Used In security_automation identity-guest-cs
No license file was found, but licenses were detected in source scan.

// Licensed under the Apache License, Version 2.0 (the "License");
// you may not use this file except in compliance with the License.
// You may obtain a copy of the License at
// http://www.apache.org/licenses/LICENSE-2.0
// distributed under the License is distributed on an "AS IS" BASIS,
/*
 * Copyright (C) 2007 The Guava Authors
 */

* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1135840621_1613612884.82/0/msgpack-core-0-8-20-sources-1-
  jar/org/msgpack/core/Preconditions.java

1.156 influxdb-java 2.20
1.156.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * The MIT License (MIT)
 */

* Copyright (c) 2017 azeti Networks AG (<info@azeti.net>)
*
* Permission is hereby granted, free of charge, to any person obtaining a copy of this software and
* associated documentation files (the "Software"), to deal in the Software without restriction,
* including without limitation the rights to use, copy, modify, merge, publish, distribute,
* sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is
* furnished to do so, subject to the following conditions:
*
* The above copyright notice and this permission notice shall be included in all copies or
* substantial portions of the Software.

* THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT
* NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR
PURPOSE AND
* NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE
FOR ANY CLAIM,
* DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE,
ARISING FROM,
* OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.
*/

Found in path(s):
* /opt/cola/permits/1140082668_1614722064.23/0/influxdb-java-2-20-sources-2-
jar/org/influxdb/impl/InfluxDBResultMapper.java
* /opt/cola/permits/1140082668_1614722064.23/0/influxdb-java-2-20-sources-2-
jar/org/influxdb/InfluxDBMapperException.java
* /opt/cola/permits/1140082668_1614722064.23/0/influxdb-java-2-20-sources-2-
jar/org/influxdb/annotation/Column.java
* /opt/cola/permits/1140082668_1614722064.23/0/influxdb-java-2-20-sources-2-
jar/org/influxdb/annotation/Measurement.java

No license file was found, but licenses were detected in source scan.

<name>The MIT License (MIT)</name>

Found in path(s):
* /opt/cola/permits/1140082668_1614722064.23/0/influxdb-java-2-20-sources-2-jar/META-
INF/maven/org.influxdb/influxdb-java/pom.xml

1.157 netty-transport-udt 4.1.69.Final

1.157.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2012 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1235364887_1638382305.22/0/netty-transport-udt-4-1-69-final-sources-
  jar/io/netty/channel/udt/UdtChannel.java
* /opt/cola/permits/1235364887_1638382305.22/0/netty-transport-udt-4-1-69-final-sources-
  jar/io/netty/channel/udt/nio/NioUdtByteConnectorChannel.java
* /opt/cola/permits/1235364887_1638382305.22/0/netty-transport-udt-4-1-69-final-sources-
  jar/io/netty/channel/udt/nio/NioUdtByteAcceptorChannel.java
* /opt/cola/permits/1235364887_1638382305.22/0/netty-transport-udt-4-1-69-final-sources-
  jar/io/netty/channel/udt/nio/NioUdtProvider.java
* /opt/cola/permits/1235364887_1638382305.22/0/netty-transport-udt-4-1-69-final-sources-
  jar/io/netty/channel/udt/nio/NioUdtMessageRendezvousChannel.java
* /opt/cola/permits/1235364887_1638382305.22/0/netty-transport-udt-4-1-69-final-sources-
  jar/io/netty/channel/udt/nio/NioUdtMessageConnectorChannel.java
* /opt/cola/permits/1235364887_1638382305.22/0/netty-transport-udt-4-1-69-final-sources-
  jar/io/netty/channel/udt/nio/NioUdtAcceptorChannel.java
* /opt/cola/permits/1235364887_1638382305.22/0/netty-transport-udt-4-1-69-final-sources-
  jar/io/netty/channel/udt/nio/NioUdtAcceptChannel.java
* /opt/cola/permits/1235364887_1638382305.22/0/netty-transport-udt-4-1-69-final-sources-
  jar/io/netty/channel/udt/nio/NioUdtMessageAcceptChannel.java
* /opt/cola/permits/1235364887_1638382305.22/0/netty-transport-udt-4-1-69-final-sources-
  jar/io/netty/channel/udt/nio/NioUdtByteRendezvousChannel.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2013 The Netty Project
 * 
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * 
 * https://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1235364887_1638382305.22/0/netty-transport-udt-4-1-69-final-sources-
  jar/io/netty/channel/udt/DefaultUdtServerChannelConfig.java
* /opt/cola/permits/1235364887_1638382305.22/0/netty-transport-udt-4-1-69-final-sources-
  jar/io/netty/channel/udt/DefaultUdtServerChannelConfig.java
/*
 * Copyright 2013 The Netty Project
 * 
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * 
 * https://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1235364887_1638382305.22/0/netty-transport-udt-4-1-69-final-sources-jar/io/netty/channel/udt/UdtServerChannel.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2012 The Netty Project
 * 
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * 
 * https://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1235364887_1638382305.22/0/netty-transport-udt-4-1-69-final-sources-jar/io/netty/channel/udt/UdtChannelConfig.java
* /opt/cola/permits/1235364887_1638382305.22/0/netty-transport-udt-4-1-69-final-sources-jar/io/netty/channel/udt/UdtServerChannel.java
* /opt/cola/permits/1235364887_1638382305.22/0/netty-transport-udt-4-1-69-final-sources-jar/io/netty/channel/udt/package-info.java
* /opt/cola/permits/1235364887_1638382305.22/0/netty-transport-udt-4-1-69-final-sources-jar/io/netty/channel/udt/DefaultUdtChannelConfig.java
No license file was found, but licenses were detected in source scan.

~ Copyright 2012 The Netty Project
~
~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~
~ https://www.apache.org/licenses/LICENSE-2.0
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.

1.158 scim2-sdk-common 2.1.1
1.158.1 Available under license :
Found license 'GNU Lesser General Public License' in '* This program is free software; you can redistribute it
and/or modify * it under the terms of the GNU General Public License (GPLv2 only) * or the terms of the GNU
Lesser General Public License (LGPlv2.1 only) * as published by the Free Software Foundation. * This program is
distributed in the hope that it will be useful, * but WITHOUT ANY WARRANTY; without even the implied
warranty of * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the * GNU General
Public License for more details. * You should have received a copy of the GNU General Public License * along
with this program; if not, see <http://www.gnu.org/licenses/>.'

1.159 hibernate 5.4.32.Final
1.159.1 Available under license :
This product bundles ASM 9.1, which is available under a "3-clause BSD"
license. For details, see licenses/ASM. For more information visit https://asm.ow2.io.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.1

1. Definitions.

1.1. "Contributor" means each individual or entity that creates or contributes to the creation of Modifications.

1.2. "Contributor Version" means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. "Covered Software" means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. "Executable" means the Covered Software in any form other than Source Code.

1.5. "Initial Developer" means the individual or entity that first
makes Original Software available under this License.

1.6. "Larger Work" means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. "License" means this document.

1.8. "Licensable" means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. "Modifications" means the Source Code and Executable form of any of the following:

A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

B. Any new file that contains any part of the Original Software or previous Modification; or

C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. "Original Software" means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. "Patent Claims" means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. "Source Code" means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. "You" (or "Your") means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, "You" includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.
2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such
(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in

Source Code form that alters or restricts the applicable version of this License or the recipients’ rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient’s rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.


You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.

Oracle is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.
You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as "Participant") alleging that the
Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. If You assert a patent infringement claim against Participant alleging that the Participant Software directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

6.4. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a "commercial item," as that term is defined
in 48 C.F.R. 2.101 (Oct. 1995), consisting of "commercial computer software" (as that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and "commercial computer software documentation" as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

------------------------------------------------------------------------
NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)
The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor
Boston, MA 02110-1335
USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and
(2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification"). Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.
2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:
a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all
its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.
9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs
If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1335 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.
signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

#

Certain source files distributed by Oracle America, Inc. and/or its affiliates are subject to the following clarification and special exception to the GPLv2, based on the GNU Project exception for its Classpath libraries, known as the GNU Classpath Exception, but only where Oracle has expressly included in the particular source file's header the words "Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the LICENSE file that accompanied this code."

You should also note that Oracle includes multiple, independent programs in this software package. Some of those programs are provided under licenses deemed incompatible with the GPLv2 by the Free Software Foundation and others. For example, the package includes programs licensed under the Apache License, Version 2.0. Such programs are licensed to you under their original licenses.

Oracle facilitates your further distribution of this package by adding the Classpath Exception to the necessary parts of its GPLv2 code, which permits you to use that code in combination with other independent modules not licensed under the GPLv2. However, note that this would not permit you to commingle code under an incompatible license with Oracle's GPLv2 licensed code by, for example, cutting and pasting such code into a file also containing Oracle's GPLv2 licensed code and then distributing the result. Additionally, if you were to remove the Classpath Exception from any of the files to which it applies and distribute the result, you would likely be required to license some or all of the other code in that distribution under the GPLv2 as well, and since the GPLv2 is incompatible with the license terms of some items included in the distribution by Oracle, removing the Classpath Exception could therefore effectively compromise your ability to further distribute the package.

Proceed with caution and we recommend that you obtain the advice of a lawyer skilled in open source matters before removing the Classpath Exception or making modifications to this package which may subsequently be redistributed and/or involve the use of third party
CLASSPATH EXCEPTION

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

This file is part of Hibernate Spatial, an extension to the hibernate ORM solution for spatial (geographic) data.

Copyright 2007-2013 Geovise BVBA

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA
(c) 2008, Adam Warski, JBoss Inc.
Java ClassMate library was originally written by Tatu Saloranta (tatu.saloranta@iki.fi)

Other developers who have contributed code are:

* Brian Langel

Checker Framework qualifiers
Copyright 2004-present by the Checker Framework developers

MIT License:
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

This copy of Java ClassMate library is licensed under Apache (Software) License, version 2.0 ("the License").

See the License for details about distribution rights, and the specific rights regarding derivate works.

You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.0

1. Definitions.

1.1. Contributor. means each individual or entity that creates or contributes to the creation of Modifications.

1.2. Contributor Version. means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. Covered Software. means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. Executable. means the Covered Software in any form other than Source Code.

1.5. Initial Developer. means the individual or entity that first makes Original Software available under this License.

1.6. Larger Work. means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. License. means this document.

1.8. Licensable. means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.
1.9. Modifications. means the Source Code and Executable form of any of the following:

A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

B. Any new file that contains any part of the Original Software or previous Modification; or

C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. Original Software. means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. Patent Claims. means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code. means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. You. (or .Your.) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, .You. includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, .control. means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.
Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients' rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However,
you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.
You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient’s rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.
Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.
You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.
When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABILITY, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND
6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as "Participant") alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY’S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a "commercial item", as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of "commercial computer software" (as that term is defined at 48 C.F.R. ? 252.227-7014(a)(1)) and "commercial computer software documentation" as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire
Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction.s conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys. fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)

The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc. 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the
software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and
disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other
system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS
How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.

Copyright (C)

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.
"CLASSPATH" EXCEPTION TO THE GPL VERSION 2

Certain source files distributed by Sun Microsystems, Inc. are subject to the following clarification and special exception to the GPL Version 2, but only where Sun has expressly included in the particular source file's header the words

"Sun designates this particular file as subject to the "Classpath" exception as provided by Sun in the License file that accompanied this code."

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License Version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

PostGIS Licensing
==================

This file attempts to include all licenses that apply within the PostGIS source tree, in particular any that are supposed to be exposed to the end user for credit requirements for instance.

PostGIS General
---------------

In general PostGIS may be distributed and/or modified under the conditions of the GNU General Public License, either version 2 or (at your option) any later version. Please refer to the COPYING file for details.

PostGIS has dependencies on other libraries which are not under GPL and that are commonly distributed with PostGIS core libraries. In general these are dynamically linked libraries. Libraries are as follows:

Proj4 - http://proj.osgeo.org -- X/MIT License
GEOS - http://geos.osgeo.org -- LGPL License
LibXML - http://xmlsoft.org/ -- X/MIT License
GDAL - http://gdal.osgeo.org/ -- X/MIT Style License

Source files included with PostGIS not under GPL
---------------------------------------------

-- The following loader/dumper files are derivative works or direct ports
of ShapeLib which is under an X/MIT License: loader/dbfopen, safileio.*, shapefil.h, shpopen.c (copyrighted by Frank Warmerdam and Even Rouault).

-- loader/getopt.* are under AT&T public domain (Copyright (c) 1984, 1985 AT&T).

-- Doc/xsl files
These are under a BSD Style license

-- liblwgeom/lookup3.c is released under Public Domain by Bob Jenkins, May 2006.

-- deps/ryu (see LICENSE in folder) is released under the Apache 2.0 license; alternatively, in can be used under the Boost 1.0 license. Copyrighted by Ulf Adams and contributors.

-- deps/uthash (see LICENSE in folder) is released under a simplified BSD license. Copyright (c) 2005-2018, Troy D. Hanson.

-- deps/wagyu (see LICENSE.README in folder) includes code released under the ISC license (copyrighted by Mapbox) and the Boost Software License (copyrighted by Angus Johnson and Mapbox). Copyright for "almost_equal.hpp" file is held by Google Inc and its license is listed at the top of that file.

The documentation for PostGIS is under a creative commons share-alike 3.0 license. http://creativecommons.org/licenses/by-sa/3.0/

Data used in documentation falls in one of the following categories
- Many of the examples in the topology and tiger geocoder section utilize data from US Census Tiger data 2010
- data or snapshots generated by community
- many images autogenerated by PostGIS documentation generator (which utilizes ImageMagick) as described
- PostGIS raster output functions such as the ST_AsPNG etc.

COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.0

1. Definitions.

1.1. Contributor. means each individual or entity that creates or contributes to the creation of Modifications.

1.2. Contributor Version. means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. Covered Software. means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. Executable. means the Covered Software in any form other than Source Code.
1.5. Initial Developer. means the individual or entity that first makes Original Software available under this License.

1.6. Larger Work. means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. License. means this document.

1.8. Licensable. means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. Modifications. means the Source Code and Executable form of any of the following:

A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

B. Any new file that contains any part of the Original Software or previous Modification; or

C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. Original Software. means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. Patent Claims. means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code. means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. You. (or .Your.) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, .You. includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, .control. means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made,
use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.
3.3. Required Notices.
You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.
You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients’ rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.
You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient’s rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.
Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.
You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.
When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN .AS IS. BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as Participant,) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING
8. U.S. GOVERNMENT END USERS.

The Covered Software is a .commercial item., as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of .commercial computer software. (as that term is defined at 48 C.F.R. ? 252.227-7014(a)(1)) and .commercial computer software documentation. as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction’s conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys’ fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)

The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.

The GNU General Public License (GPL) Version 2, June 1991
Copyright (C) 1989, 1991 Free Software Foundation, Inc. 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under
copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification"). Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.
3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-
free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY
COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIIBUTE THE
PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL,
SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY
TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING
RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF
THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER
PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to
achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to
most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a
pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.

Copyright (C)

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public
License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later
version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the
implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General
Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the
Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and
you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License.
Of course, the commands you use may be called something other than `show w' and `show c'; they could even be
mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright
Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

"CLASSPATH" EXCEPTION TO THE GPL VERSION 2

Certain source files distributed by Sun Microsystems, Inc. are subject to the following clarification and special exception to the GPL Version 2, but only where Sun has expressly included in the particular source file's header the words

"Sun designates this particular file as subject to the "Classpath" exception as provided by Sun in the License file that accompanied this code."

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License Version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.
Copyright 2019, OnGres.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR
ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES 
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; 
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON 
ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT 
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS 
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

ASM: a very small and fast Java bytecode manipulation framework
Copyright (c) 2000-2011 INRIA, France Telecom
All rights reserved.

Redistribution and use in source and binary forms, with or without 
modification, are permitted provided that the following conditions 
are met:
1. Redistributions of source code must retain the above copyright 
notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright 
notice, this list of conditions and the following disclaimer in the 
documentation and/or other materials provided with the distribution.
3. Neither the name of the copyright holders nor the names of its 
contributors may be used to endorse or promote products derived from 
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" 
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE 
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE 
ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE 
LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR 
CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF 
SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS 
INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN 
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) 
ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF 
THE POSSIBILITY OF SUCH DAMAGE.

# The main CI of Hibernate ORM is https://ci.hibernate.org/job/hibernate-orm-6.0-h2-main/.
# However, Hibernate ORM builds run on GitHub actions regularly 
# to check that it still works and can be used in GitHub forks.
# See https://docs.github.com/en/free-pro-team@latest/actions 
# for more information about GitHub actions.

name: Hibernate ORM build

on:
push:
  branches: 
    - ‘5.4’
pull_request:
  branches: 
    - ‘5.4’
jobs:
build:
  name: Java 8
  runs-on: ubuntu-latest
  # We want to know the test results of all matrix entries
  continue-on-error: true
  strategy:
    fail-fast: false
  matrix:
    # When GitHub Actions supports it: https://github.com/actions/toolkit/issues/399
    # We will use the experimental flag as indicator whether a failure should cause a workflow failure
    include:
      - rdbms: h2
        experimental: false
      # - rdbms: derby
      #   experimental: true
      # - rdbms: mariadb
      #   experimental: true
      - rdbms: postgresql
        experimental: true
      # - rdbms: oracle
      #   experimental: true
      # - rdbms: db2
      #   experimental: true
      # - rdbms: mssql
      #   experimental: true
  steps:
    - uses: actions/checkout@v2
    - name: Reclaim Disk Space
      run: .github/ci-prerequisites.sh
    - name: Set up Java 8
      uses: actions/setup-java@v1
      with:
        java-version: 1.8
    - name: Get year/month for cache key
      id: get-date
      run: |
        echo "::set-output name=yearmonth::$(/bin/date -u +%Y-%m)"
      shell: bash
    - name: Cache Maven local repository
      uses: actions/cache@v2
      id: cache-maven
      with:
        path: |
        ~/.m2/repository
        ~/.gradle/caches/
        ~/.gradle/wrapper/
        # refresh cache every month to avoid unlimited growth
key: maven-localrepo-$\{$ steps.get-date.outputs.yearmonth \}$
- name: Run build script
  env:
    RDBMS: $\{$ matrix.rdbms \}$
  run: ./ci/build-github.sh
  shell: bash
- name: Upload test reports (if Gradle failed)
  uses: actions/upload-artifact@v2
  if: failure()
  with:
    name: test-reports-java8-$\{$ matrix.rdbms \}$
    path: |
      ./**/target/reports/tests/
      ./**/target/reports/checkstyle/
- name: Omit produced artifacts from build cache
  run: ./ci/before-cache.sh

build11:
  name: Java 11
  runs-on: ubuntu-latest
  # We want to know the test results of all matrix entries
  continue-on-error: true
  steps:
    - uses: actions/checkout@v2
    - name: Set up Java 11
      uses: actions/setup-java@v1
      with:
        java-version: 11
    - name: Get year/month for cache key
      id: get-date
      run: |
        echo "::set-output name=yearmonth::$(/bin/date -u "%Y-%m")"
      shell: bash
    - name: Cache Maven local repository
      uses: actions/cache@v2
      id: cache-maven
      with:
        path: |
          ~/.m2/repository
          ~/.gradle/caches/
          ~/.gradle/wrapper/
      # refresh cache every month to avoid unlimited growth
      key: maven-localrepo-$\{$ steps.get-date.outputs.yearmonth \}$
    - name: Run build script
      run: ./ci/build-github.sh
      shell: bash
    - name: Upload test reports (if Gradle failed)
      uses: actions/upload-artifact@v2
      if: failure()
with:
name: test-reports-java11
path: |
  ./**/target/reports/tests/
  ./**/target/reports/checkstyle/
- name: Omit produced artifacts from build cache
  run: ./ci/before-cache.sh

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA  02110-1301  USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts
as the successor of the GNU Library Public License, version 2, hence
the version number 2.1.]

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
Licenses are intended to guarantee your freedom to share and change
free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some
specially designated software packages--typically libraries--of the
Free Software Foundation and other authors who decide to use it. You
can use it too, but we suggest you first think carefully about whether
this license or the ordinary General Public License is the better
strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use,
not price. Our General Public Licenses are designed to make sure that
you have the freedom to distribute copies of free software (and charge
for this service if you wish); that you receive source code or can get
it if you want it; that you can change the software and use pieces of
it in new free programs; and that you are informed that you can do
these things.

To protect your rights, we need to make restrictions that forbid
distributors to deny you these rights or to ask you to surrender these
rights. These restrictions translate to certain responsibilities for
you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis
or for a fee, you must give the recipients all the rights that we gave
you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to
encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means
all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any

---
application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.
If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the
copyright notice for the Library among them, as well as a reference
directing the user to the copy of this License. Also, you must do one
of these things:

a) Accompany the work with the complete corresponding
machine-readable source code for the Library including whatever
changes were used in the work (which must be distributed under
Sections 1 and 2 above); and, if the work is an executable linked
with the Library, with the complete machine-readable "work that
uses the Library", as object code and/or source code, so that the
user can modify the Library and then relink to produce a modified
executable containing the modified Library. (It is understood
that the user who changes the contents of definitions files in the
Library will not necessarily be able to recompile the application
to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the
Library. A suitable mechanism is one that (1) uses at run time a
copy of the library already present on the user's computer system,
rather than copying library functions into the executable, and (2)
will operate properly with a modified version of the library, if
the user installs one, as long as the modified version is
interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at
least three years, to give the same user the materials
specified in Subsection 6a, above, for a charge no more
than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy
from a designated place, offer equivalent access to copy the above
specified materials from the same place.

e) Verify that the user has already received a copy of these
materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the
Library" must include any data and utility programs needed for
reproducing the executable from it. However, as a special exception,
the materials to be distributed need not include anything that is
normally distributed (in either source or binary form) with the major
components (compiler, kernel, and so on) of the operating system on
which the executable runs, unless that component itself accompanies
the executable.

It may happen that this requirement contradicts the license
restrictions of other proprietary libraries that do not normally
accompany the operating system. Such a contradiction means you cannot
use both them and the Library together in an executable that you
distribute.

7. You may place library facilities that are a work based on the
Library side-by-side in a single library together with other library
facilities not covered by this License, and distribute such a combined
library, provided that the separate distribution of the work based on
the Library and of the other library facilities is otherwise
permitted, and provided that you do these two things:

   a) Accompany the combined library with a copy of the same work
      based on the Library, uncombined with any other library
      facilities. This must be distributed under the terms of the
      Sections above.

   b) Give prominent notice with the combined library of the fact
      that part of it is a work based on the Library, and explaining
      where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute
the Library except as expressly provided under this License. Any
attempt otherwise to copy, modify, sublicense, link with, or
distribute the Library is void, and will automatically terminate your
rights under this License. However, parties who have received copies,
or rights, from you under this License will not have their licenses
terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not
signed it. However, nothing else grants you permission to modify or
distribute the Library or its derivative works. These actions are
prohibited by law if you do not accept this License. Therefore, by
modifying or distributing the Library (or any work based on the
Library), you indicate your acceptance of this License to do so, and
all its terms and conditions for copying, distributing or modifying
the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the
Library), the recipient automatically receives a license from the
original licensor to copy, distribute, link with or modify the Library
subject to these terms and conditions. You may not impose any further
restrictions on the recipients' exercise of the rights granted herein.
You are not responsible for enforcing compliance by third parties with
this License.

11. If, as a consequence of a court judgment or allegation of patent
infringement or for any other reason (not limited to patent issues),
conditions are imposed on you (whether by court order, agreement or
otherwise) that contradict the conditions of this License, they do not
excuse you from the conditions of this License. If you cannot
distribute so as to satisfy simultaneously your obligations under this
License and any other pertinent obligations, then as a consequence you
may not distribute the Library at all. For example, if a patent
license would not permit royalty-free redistribution of the Library by
all those who receive copies directly or indirectly through you, then
the only way you could satisfy both it and this License would be to
refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any
particular circumstance, the balance of the section is intended to apply,
and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any
patents or other property right claims or to contest validity of any
such claims; this section has the sole purpose of protecting the
integrity of the free software distribution system which is
implemented by public license practices. Many people have made
generous contributions to the wide range of software distributed
through that system in reliance on consistent application of that
system; it is up to the author/donor to decide if he or she is willing
to distribute software through any other system and a licensee cannot
impose that choice.

This section is intended to make thoroughly clear what is believed to
be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in
certain countries either by patents or by copyrighted interfaces, the
original copyright holder who places the Library under this License may add
an explicit geographical distribution limitation excluding those countries,
so that distribution is permitted only in or among countries not thus
excluded. In such case, this License incorporates the limitation as if
written in the body of this License.

13. The Free Software Foundation may publish revised and/or new
versions of the Lesser General Public License from time to time.
Such new versions will be similar in spirit to the present version,
but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library
specifies a version number of this License which applies to it and
“any later version”, you have the option of following the terms and
conditions either of that version or of any later version published by
the Free Software Foundation. If the Library does not specify a
license version number, you may choose any version ever published by
the Free Software Foundation.
14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year>  <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!

COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)Version 1.1

1. Definitions.

1.1. "Contributor" means each individual or entity that creates or contributes to the creation of Modifications.

1.2. "Contributor Version" means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. "Covered Software" means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. "Executable" means the Covered Software in any form other than Source Code.

1.5. "Initial Developer" means the individual or entity that first makes Original Software available under this License.

1.6. "Larger Work" means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.
1.7. "License" means this document.

1.8. "Licensable" means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. "Modifications" means the Source Code and Executable form of any of the following:

A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

B. Any new file that contains any part of the Original Software or previous Modification; or

C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. "Original Software" means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. "Patent Claims" means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. "Source Code" means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. "You" (or "Your") means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, "You" includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the
Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the
Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients' rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.


You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.

Oracle is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.
4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXPECT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as "Participant") alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. If You assert a patent infringement claim against Participant alleging that the Participant Software directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

6.4. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING
NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a "commercial item," as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of "commercial computer software" (as that term is defined at 48 C.F.R. ? 252.227-7014(a)(1)) and "commercial computer software documentation" as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

--------
NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)
The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-
law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc. 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.
The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on
the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives
a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE
PROGRAM. TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.

Copyright (C)

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:
Gnomovision version 69, Copyright (C) year name of author

Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

"CLASSPATH" EXCEPTION TO THE GPL VERSION 2

Certain source files distributed by Oracle are subject to the following clarification and special exception to the GPL Version 2, but only where Oracle has expressly included in the particular source file's header the words "Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the License file that accompanied this code."

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License Version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

Copyright 2014 - ${current.year} Rafael Winterhalter

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts as the successor of the GNU Library Public License, version 2, hence the version number 2.1.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them
We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be
allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated
interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

   You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) The modified work must itself be a software library.

   b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

   c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

   d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

   (For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any
application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.
If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by
this License. You must supply a copy of this License. If the work
during execution displays copyright notices, you must include the
copyright notice for the Library among them, as well as a reference
directing the user to the copy of this License. Also, you must do one
of these things:

a) Accompany the work with the complete corresponding
machine-readable source code for the Library including whatever
changes were used in the work (which must be distributed under
Sections 1 and 2 above); and, if the work is an executable linked
with the Library, with the complete machine-readable "work that
uses the Library", as object code and/or source code, so that the
user can modify the Library and then relink to produce a modified
executable containing the modified Library. (It is understood
that the user who changes the contents of definitions files in the
Library will not necessarily be able to recompile the application
to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the
Library. A suitable mechanism is one that (1) uses at run time a
copy of the library already present on the user's computer system,
rather than copying library functions into the executable, and (2)
will operate properly with a modified version of the library, if
the user installs one, as long as the modified version is
interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at
least three years, to give the same user the materials
specified in Subsection 6a, above, for a charge no more
than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy
from a designated place, offer equivalent access to copy the above
specified materials from the same place.

e) Verify that the user has already received a copy of these
materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the
Library" must include any data and utility programs needed for
reproducing the executable from it. However, as a special exception,
the materials to be distributed need not include anything that is
normally distributed (in either source or binary form) with the major
components (compiler, kernel, and so on) of the operating system on
which the executable runs, unless that component itself accompanies
the executable.

It may happen that this requirement contradicts the license
restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.
11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by
the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS
Copyright 2017, OnGres.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or
other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright (c) 1997, PostgreSQL Global Development Group
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Additional License files can be found in the 'licenses' folder located in the same directory as the LICENSE file (i.e. this file)

- Software produced outside the ASF which is available under other licenses (not Apache-2.0)

BSD-2-Clause
* com.ongres.scram:client:2.1
* com.ongres.scram:common:2.1
* com.ongres.stringprep:saslprep:1.1
* com.ongres.stringprep:stringprep:1.1

Apache License
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

Open Source Used In security_automation identity-guest-cs 976
http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.160 spring-aspects 5.3.12

1.160.1 Available under license:

No license file was found, but licenses were detected in source scan.

/ *
* Copyright 2002-2012 the original author or authors.
* *
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* *
*  https://www.apache.org/licenses/LICENSE-2.0
* *
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
  */opt/cola/permits/1221016774_1635406660.16/0/spring-aspects-5-3-12-sources.jar/org/springframework/context/annotation/aspectj/EnableSpringConfigured.java

No license file was found, but licenses were detected in source scan.

/ *
* Copyright 2002-2020 the original author or authors.
* *
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* *
*  https://www.apache.org/licenses/LICENSE-2.0
* *
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1221016774_1635406660.16/0/spring-aspects-5-3-12-sources-jar/org/springframework/scheduling/aspectj/AspectJAsyncConfiguration.java
* /opt/cola/permits/1221016774_1635406660.16/0/spring-aspects-5-3-12-sources-jar/org/springframework/beans/factory/aspectj/AnnotationBeanConfigurerAspect.aj
* /opt/cola/permits/1221016774_1635406660.16/0/spring-aspects-5-3-12-sources-jar/org/springframework/beans/factory/aspectj/AbstractInterfaceDrivenDependencyInjectionAspect.aj
* /opt/cola/permits/1221016774_1635406660.16/0/spring-aspects-5-3-12-sources-jar/org/springframework/cache/aspectj/AspectJJCacheConfiguration.java
* /opt/cola/permits/1221016774_1635406660.16/0/spring-aspects-5-3-12-sources-jar/org/springframework/transaction/aspectj/AspectJTransactionManagementConfiguration.java
* /opt/cola/permits/1221016774_1635406660.16/0/spring-aspects-5-3-12-sources-jar/org/springframework/transaction/aspectj/AbstractTransactionAspect.aj
* /opt/cola/permits/1221016774_1635406660.16/0/spring-aspects-5-3-12-sources-jar/org/springframework/cache/aspectj/AspectJCachingConfiguration.java
* /opt/cola/permits/1221016774_1635406660.16/0/spring-aspects-5-3-12-sources-jar/org/springframework/cache/aspectj/AspectJJtaTransactionManagementConfiguration.java

No license file was found, but licenses were detected in source scan.

/ *
* Copyright 2002-2021 the original author or authors.
* *
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* *
*   https://www.apache.org/licenses/LICENSE-2.0
 *
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1221016774_1635406660.16/0/spring-aspects-5-3-12-sources-jar/org/springframework/beans/factory/aspectj/ConfigurableObject.java

No license file was found, but licenses were detected in source scan.

/ *
* Copyright 2002-2013 the original author or authors.
* *
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*  
*         https://www.apache.org/licenses/LICENSE-2.0
*  
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1221016774_1635406660.16/0/spring-aspects-5-3-12-sources-
jar/org/springframework/cache/aspectj/AnnotationCacheAspect.aj
No license file was found, but licenses were detected in source scan.

/"
* Copyright 2002-2017 the original author or authors.
*  
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*  
*         https://www.apache.org/licenses/LICENSE-2.0
*  
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1221016774_1635406660.16/0/spring-aspects-5-3-12-sources-
jar/org/springframework/beans/factory/aspectj/GenericInterfaceDrivenDependencyInjectionAspect.aj
No license file was found, but licenses were detected in source scan.

/"
* Copyright 2002-2014 the original author or authors.
*  
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*  
*         https://www.apache.org/licenses/LICENSE-2.0
*  
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1221016774_1635406660.16/0/spring-aspects-5-3-12-sources-jar/org/springframework/beans/factory/aspectj/AbstractDependencyInjectionAspect.aj
* /opt/cola/permits/1221016774_1635406660.16/0/spring-aspects-5-3-12-sources-jar/org/springframework/cache/aspectj/JCacheCacheAspect.aj
* /opt/cola/permits/1221016774_1635406660.16/0/spring-aspects-5-3-12-sources-jar/org/springframework/cache/aspectj/AbstractCacheAspect.aj
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2002-2016 the original author or authors.
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1221016774_1635406660.16/0/spring-aspects-5-3-12-sources-jar/org/springframework/scheduling/aspectj/AbstractAsyncExecutionAspect.aj
* /opt/cola/permits/1221016774_1635406660.16/0/spring-aspects-5-3-12-sources-jar/org/springframework/scheduling/aspectj/AnnotationAsyncExecutionAspect.aj
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2002-2015 the original author or authors.
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
*/opt/cola/permits/1221016774_1635406660.16/0/spring-aspects-5-3-12-sources-jar/org/springframework/transaction/aspectj/AnnotationTransactionAspect.aj
*/opt/cola/permits/1221016774_1635406660.16/0/spring-aspects-5-3-12-sources-jar/org/springframework/transaction/aspectj/JtaAnnotationTransactionAspect.aj
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2002-2018 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
*/opt/cola/permits/1221016774_1635406660.16/0/spring-aspects-5-3-12-sources-jar/org/springframework/cache/aspectj/AnyThrow.java
*/opt/cola/permits/1221016774_1635406660.16/0/spring-aspects-5-3-12-sources-jar/org/springframework/context/annotation/aspectj/SpringConfiguredConfiguration.java

1.161 mime-pull 1.9.15
1.161.1 Available under license:

[[/]]: # " Copyright (c) 2018, 2021 Oracle and/or its affiliates. All rights reserved. 
[[/]]: # "
[[/]]: # " This program and the accompanying materials are made available under the "
[[/]]: # " terms of the Eclipse Distribution License v. 1.0, which is available at "
[[/]]: # "
[[/]]: # " SPDX-License-Identifier: BSD-3-Clause "

Copyright (c) 2018 Oracle and/or its affiliates. All rights reserved.
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

- Neither the name of the Eclipse Foundation, Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

# Notices for Eclipse Metro

This content is produced and maintained by the Eclipse Metro project.

* Project home: https://projects.eclipse.org/projects/ee4j.metro

## Trademarks

Eclipse Metro is a trademark of the Eclipse Foundation.

## Copyright

All content is the property of the respective authors or their employers. For
more information regarding authorship of content, please consult the listed source code repository logs.

## Declared Project Licenses

This program and the accompanying materials are made available under the terms of the Eclipse Distribution License v. 1.0 which is available at http://www.eclipse.org/org/documents/edl-v10.php.

SPDX-License-Identifier: BSD-3-Clause

## Source Code

The project maintains the following source code repositories:

* https://github.com/eclipse-ee4j/metro-xmlstreambuffer
* https://github.com/eclipse-ee4j/metro-policy
* https://github.com/eclipse-ee4j/metro-wsit
* https://github.com/eclipse-ee4j/metro-mimepull
* https://github.com/eclipse-ee4j/metro-ws-test-harness
* https://github.com/eclipse-ee4j/metro-package-rename-task
* https://github.com/eclipse-ee4j/metro-jax-ws
* https://github.com/eclipse-ee4j/metro-saaj
* https://github.com/eclipse-ee4j/metro-jwsp-samples
* https://github.com/eclipse-ee4j/jax-rpc-ri

## Third-party Content

This project leverages the following third party content.

addressing.xml Version: 2004/10 (n/a)

* License: W3C
* Project: https://www.w3.org/Submission/ws-addressing/
* Source: http://schemas.xmlsoap.org/ws/2004/08/addressing/

ant-launcher (1.10.2)

* License: Apache-2.0 AND SAX-PD AND W3C
* Project: https://ant.apache.org/
* Source: http://central.maven.org/maven2/ant/ant-launcher/1.10.2/ant-launcher-1.10.2-sources.jar

Apache Ant (1.6)

* License: Apache-1.1
* Project: https://ant.apache.org/
* Source: https://repo1.maven.org/maven2/ant/ant/1.6/ant-1.6-sources.jar
Apache Ant (1.10.2)

* License: Apache-2.0 AND W3C AND LicenseRef-Public-Domain

commons-loging (1.1.2)

* License: Apache-2.0
* Project: https://commons.apache.org/proper/commons-loging/
* Source:
  http://central.maven.org/maven2/commons-loging/commons-loging/1.1.2/commons-loging-1.1.2-sources.jar

JUnit (4.12)

* License: Eclipse Public License

maven-core (3.5.2)

* License: Apache-2.0

maven-plugin-annotations (3.5.1)

* License: Apache-2.0
* Project:
* Source:
  https://github.com/apache/maven-plugin-tools/tree/maven-plugin-tools-3.5.1/maven-plugin-annotations

maven-plugin-api (3.5.2)

* License: Apache-2.0
* Project: https://maven.apache.org/
* Source: https://github.com/apache/maven/tree/master/maven-plugin-api

maven-resolver-api (1.1.1)

* License: Apache-2.0

maven-resolver-util (1.1.1)

* License: Apache-2.0

maven-settings (3.5.2)

* License: Apache-2.0

mex.xsd Version: 2004/09 (n/a)
* License: Oasis Style
* Project: https://www.w3.org/Submission/WS-MetadataExchange/#appendix-II

plexus-utils (3.1.0)

* License: Apache-2.0 or Apache-1.1 or BSD or Public Domain or Indiana University Extreme! Lab Software License V1.1.1 (Apache 1.1 style)

relaxng-datatype (1.0)

* License: New BSD license

stax2-api (4.1)

* License: Pending
* Project: https://github.com/FasterXML/stax2-api
testng (6.14.2)

* License: Apache-2.0 AND MIT
* Project: https://testng.org/doc/index.html
* Source: https://github.com/cbeust/testng

woodstox-core-asl (4.4.1)

* License: Apache-2.0

woodstox-core-asl (5.1.0)

* License: Pending
* Project: https://github.com/FasterXML/woodstox
* Source: https://github.com/FasterXML/woodstox

ws-addr.wsd (1.0)

* License: W3C
* Project: https://www.w3.org/2005/08/addressing/
* Source: https://www.w3.org/2006/03/addressing/ws-addr.xsd

wsat.xsd Version: 2004/10 (n/a)

* License: Oasis Style
* Source: http://schemas.xmlsoap.org/ws/2004/10/wsat/wsat.xsd
wscoor.xsd (1.0)

* License: OASIS Style

wscoor.xsd (1.1)

* License: Oasis (Custom)
* Project: http://docs.oasis-open.org/ws-tx/wscoor/2006/06
* Source:
  http://docs.oasis-open.org/ws-tx/wscoor/2006/06/wstx-wscoor-1.1-schema-200701.xsd

wsrm Version: 2005/02 (n/a)

* License: Oasis (Custom)
* Project: http://schemas.xmlsoap.org/ws/2005/02/rm/
* Source:

wsrm.xsd (1.2)

* License: Oasis

wstx-wsat.xsd (1.1)

* License: Oasis (Custom)

xmlsec (1.5.8)

* License: Apache-2.0
* Project: http://santuario.apache.org/
* Source:
  https://repo1.maven.org/maven2/org/apache/santuario/xmlsec/1.5.8/xmlsec-1.5.8-sources.jar

## Cryptography

Content may contain encryption software. The country in which you are currently may have restrictions on the import, possession, and use, and/or re-export to another country, of encryption software. BEFORE using any encryption software, please check the country's laws, regulations and policies concerning the import, possession, or use, and re-export of encryption software, to see if this is permitted.

1.162 jetty-servlets 9.4.44.v20210927
1.62.1 Available under license:

This program and the accompanying materials are made available under the terms of the Eclipse Public License 2.0 which is available at http://www.eclipse.org/legal/epl-2.0, or the Apache Software License 2.0 which is available at https://www.apache.org/licenses/LICENSE-2.0.

Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

“Contribution” means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and
b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

“Contributor” means any person or entity that distributes the Program.

"Licensed Patents” mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program” means the Contributions distributed in accordance with this Agreement.

"Recipient” means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly
perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and

b) its license agreement:
   i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
   ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;
   iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and
   iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.
When the Program is made available in source code form:

a) it must be made available under this Agreement; and
b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must:

a) promptly notify the Commercial Contributor in writing of such claim, and
b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN
"AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time.
This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
that You distribute, all copyright, patent, trademark, and
attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.
You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

==============================================================
Jetty Web Container
Copyright 1995-2018 Mort Bay Consulting Pty Ltd.
==============================================================

The Jetty Web Container is Copyright Mort Bay Consulting Pty Ltd unless otherwise noted.

Jetty is dual licensed under both

* The Apache 2.0 License
  http://www.apache.org/licenses/LICENSE-2.0.html

  and

* The Eclipse Public 1.0 License
Jetty may be distributed under either license.

------

Eclipse

The following artifacts are EPL.
* org.eclipse.jetty.orbit:org.eclipse.jdt.core

The following artifacts are EPL and ASL2.
* org.eclipse.jetty.orbit:javax.security.auth.message

The following artifacts are EPL and CDDL 1.0.
* org.eclipse.jetty.orbit:javax.mail.glassfish

------

Oracle

The following artifacts are CDDL + GPLv2 with classpath exception.
https://glassfish.dev.java.net/nonav/public/CDDL/GPL.html

* javax.servlet:javax.servlet-api
* javax.annotation:javax.annotation-api
* javax.transaction:javax.transaction-api
* javax.websocket:javax.websocket-api

------

Oracle OpenJDK

If ALPN is used to negotiate HTTP/2 connections, then the following artifacts may be included in the distribution or downloaded when ALPN module is selected.

* java.sun.security.ssl

These artifacts replace/modify OpenJDK classes. The modifications are hosted at github and both modified and original are under GPL v2 with classpath exceptions.
http://openjdk.java.net/legal/gplv2+ce.html

------

OW2

The following artifacts are licensed by the OW2 Foundation according to the terms of http://asm.ow2.org/license.html
org.ow2.asm:asm-commons
org.ow2.asm:asm

-----
Apache

The following artifacts are ASL2 licensed.

org.apache.taglibs:taglibs-standard-spec
org.apache.taglibs:taglibs-standard-impl

-----
MortBay

The following artifacts are ASL2 licensed. Based on selected classes from following Apache Tomcat jars, all ASL2 licensed.

org.mortbay.jasper:apache-jsp
org.apache.tomcat:tomcat-jasper
org.apache.tomcat:tomcat-juli
org.apache.tomcat:tomcat-jsp-api
org.apache.tomcat:tomcat-el-api
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-api
org.apache.tomcat:tomcat-util-scan
org.apache.tomcat:tomcat-util

org.mortbay.jasper:apache-el
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-el-api

-----
Mortbay

The following artifacts are CDDL + GPLv2 with classpath exception.

https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

org.eclipse.jetty.toolchain:jetty-schemas

-----
Assorted

The UnixCrypt.java code implements the one way cryptography used by
Unix systems for simple password protection. Copyright 1996 Aki Yoshida, modified April 2001 by Iris Van den Broeke, Daniel Deville. Permission to use, copy, modify and distribute UnixCrypt for non-commercial or commercial purposes and without fee is granted provided that the copyright notice appears in all copies.

1.163 curator-client 5.1.0

1.163.1 Available under license:

Curator Client
Copyright 2011-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of Your accepting any such warranty or additional liability.
APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.164 log4j-to-slf4j 2.17.0

1.164.1 Available under license:

Apache Log4j to SLF4J Adapter
Copyright 1999-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.
"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "["] replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
1.165 commons-beanutils 1.8.3

1.165.1 Available under license:

Apache Commons BeanUtils
Copyright 2000-2010 The Apache Software Foundation

This product includes software developed by
The Apache Software Foundation (http://www.apache.org/).
/
* Licensed to the Apache Software Foundation (ASF) under one or more
* contributor license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright ownership.
* The ASF licenses this file to You under the Apache License, Version 2.0
* (the "License"); you may not use this file except in compliance with
* the License. You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.
"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.166 java-driver-query-builder 4.10.0

1.166.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright DataStax, Inc.
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License.
 */
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1147203698_1616863303.7/0/java-driver-query-builder-4-10-0-sources-jar/com/datastax/dse/driver/api/querybuilder/schema/CreateGraphSpaceStart.java
* /opt/cola/permits/1147203698_1616863303.7/0/java-driver-query-builder-4-10-0-sources-jar/com/datastax/dse/driver/api/querybuilder/schema/AlterGraphSpaceStart.java
* /opt/cola/permits/1147203698_1616863303.7/0/java-driver-query-builder-4-10-0-sources-jar/com/datastax/dse/driver/api/querybuilder/schema/AlterGraphPathStart.java
* /opt/cola/permits/1147203698_1616863303.7/0/java-driver-query-builder-4-10-0-sources-jar/com/datastax/dse/driver/api/querybuilder/schema/CreateFunction.java
* /opt/cola/permits/1147203698_1616863303.7/0/java-driver-query-builder-4-10-0-sources-jar/com/datastax/dse/driver/api/querybuilder/schema/CreateTable.java
* /opt/cola/permits/1147203698_1616863303.7/0/java-driver-query-builder-4-10-0-sources-jar/com/datastax/dse/driver/api/querybuilder/schema/AlterTableDropColumn.java
* /opt/cola/permits/1147203698_1616863303.70/java-driver-query-builder-4-10-0-sources.jar/com/datastax/dse/driver/internal/querybuilder/schema/DseTableEdgeOperation.java
* /opt/cola/permits/1147203698_1616863303.70/java-driver-query-builder-4-10-0-sources.jar/com/datastax/oss/driver/internal/querybuilder/schema/compaction/DefaultTimeWindowCompactionStrategy.java
* /opt/cola/permits/1147203698_1616863303.70/java-driver-query-builder-4-10-0-sources.jar/com/datastax/oss/driver/api/querybuilder/schema/CreateAggregateStart.java
* /opt/cola/permits/1147203698_1616863303.70/java-driver-query-builder-4-10-0-sources.jar/com/datastax/oss/driver/api/querybuilder/schema/AlterTableAddColumn.java
* /opt/cola/permits/1147203698_1616863303.70/java-driver-query-builder-4-10-0-sources.jar/com/datastax/oss/driver/api/querybuilder/relation/InRelationBuilder.java
* /opt/cola/permits/1147203698_1616863303.70/java-driver-query-builder-4-10-0-sources.jar/com/datastax/dse/driver/internal/querybuilder/schema/DseTableVertexOperation.java
* /opt/cola/permits/1147203698_1616863303.70/java-driver-query-builder-4-10-0-sources.jar/com/datastax/oss/driver/api/querybuilder/schema/AlterTypeRenameFieldEnd.java
* /opt/cola/permits/1147203698_1616863303.70/java-driver-query-builder-4-10-0-sources.jar/com/datastax/oss/driver/api/querybuilder/relation/DefaultTokenRelationBuilder.java
* /opt/cola/permits/1147203698_1616863303.70/java-driver-query-builder-4-10-0-sources.jar/com/datastax/oss/driver/api/querybuilder/schema/CreateAggregateEnd.java
* /opt/cola/permits/1147203698_1616863303.70/java-driver-query-builder-4-10-0-sources.jar/com/datastax/oss/driver/api/querybuilder/schema/compaction/SizeTieredCompactionStrategy.java
* /opt/cola/permits/1147203698_1616863303.70/java-driver-query-builder-4-10-0-sources.jar/com/datastax/oss/driver/api/querybuilder/schema/AlterMaterializedView.java
* /opt/cola/permits/1147203698_1616863303.70/java-driver-query-builder-4-10-0-sources.jar/com/datastax/oss/driver/api/querybuilder/schema/compaction/CompactionStrategy.java
* /opt/cola/permits/1147203698_1616863303.70/java-driver-query-builder-4-10-0-sources.jar/com/datastax/dse/driver/api/querybuilder/schema/AlterDseTableWithOptions.java
* /opt/cola/permits/1147203698_1616863303.70/java-driver-query-builder-4-10-0-sources.jar/com/datastax/oss/driver/api/querybuilder/schema/CreateIndex.java
* /opt/cola/permits/1147203698_1616863303.70/java-driver-query-builder-4-10-0-sources.jar/com/datastax/oss/driver/api/querybuilder/schema/AlterTableWithOptionsEnd.java
* /opt/cola/permits/1147203698_1616863303.70/java-driver-query-builder-4-10-0-sources.jar/com/datastax/dse/driver/api/querybuilder/schema/CreateDseFunctionWithType.java
* /opt/cola/permits/1147203698_1616863303.70/java-driver-query-builder-4-10-0-sources.jar/com/datastax/oss/driver/api/querybuilder/schema/Select.java
* /opt/cola/permits/1147203698_1616863303.70/java-driver-query-builder-4-10-0-sources.jar/com/datastax/oss/driver/api/querybuilder/relation/TokenRelationBuilder.java
* /opt/cola/permits/1147203698_1616863303.70/java-driver-query-builder-4-10-0-sources.jar/com/datastax/oss/driver/api/querybuilder/insert/RegularInsert.java
1.169 paranamer 2.7

1.170 jboss-transaction-api 1.1.1.Final
1.170.1 Available under license:

Found license 'Common Development and Distribution License (CDDL) 1.1' in '<name>Common Development and Distribution License</name> <name>GNU General Public License, Version 2 with the Classpath Exception</name>'
Found license 'General Public License 2.0' in '<name>Common Development and Distribution License</name> <name>GNU General Public License, Version 2 with the Classpath Exception</name>'

1.171 jersey 1.19.1
1.171.1 Available under license:

If you received the contents of this file pursuant to a Technology License and Distribution Agreement (TLDA) or Sun Community Source License (SCSL) agreement with Sun, then such TLDA or SCSL, whichever is applicable, and not the license found at or referenced by a License.txt (or License.html) file or a source header file, shall apply to your use of the contents of the file.

DO NOT TRANSLATE OR LOCALIZE.

%%% The following software may be included in this product: Jettison
Use of any of this software is governed by the terms of the license below:

Copyright 2006 Envoi Solutions LLC

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and limitations under the License.

--------------------------------------------------------------------------

Apache License
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to
You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

1. You must give any other recipients of the Work or Derivative Works a copy of this License; and

2. You must cause any modified files to carry prominent notices stating that You changed the files; and

3. You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

4. If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and
reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
APPENDIX: How to apply the Apache License to your work

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner] Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at http://www.apache.org/licenses/LICENSE-2.0 Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

%% The following software may be included in this product: Jackson
Use of any of this software is governed by the terms of the license below:

Jackson is dual-licensed under two alternative popular Open Source licenses: Apache (AL 2.0) and Gnu Lesser GPL (LGPL 2.1). You choose one or the other, as necessary (if you want to redistribute the code for use, you do not need license), and abide by the license rules as defined by the respective license agreement (and only that one).
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."
"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

1. You must give any other recipients of the Work or Derivative Works a copy of this License; and

2. You must cause any modified files to carry prominent notices stating that You changed the files; and

3. You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

4. If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of
any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner] Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at http://www.apache.org/licenses/LICENSE-2.0 Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

DO NOT ALTER OR REMOVE COPYRIGHT NOTICES OR THIS HEADER.

Copyright (c) 2010-2011 Oracle and/or its affiliates. All rights reserved.

The contents of this file are subject to the terms of either the GNU
General Public License Version 2 only ("GPL") or the Common Development and Distribution License ("CDDL") (collectively, the "License"). You may not use this file except in compliance with the License. You can obtain a copy of the License at http://glassfish.java.net/public/CDDL+GPL_1_1.html or packager/legal/LICENSE.txt. See the License for the specific language governing permissions and limitations under the License.

When distributing the software, include this License Header Notice in each file and include the License file at packager/legal/LICENSE.txt.

GPL Classpath Exception:
Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the GPL Version 2 section of the License file that accompanied this code.

Modifications:
If applicable, add the following below the License Header, with the fields enclosed by brackets [] replaced by your own identifying information: "Portions Copyright [year] [name of copyright owner]"

Contributor(s):
If you wish your version of this file to be governed by only the CDDL or only the GPL Version 2, indicate your decision by adding "[Contributor] elects to include this software in this distribution under the [CDDL or GPL Version 2] license." If you don't indicate a single choice of license, a recipient has the option to distribute your version of this file under either the CDDL, the GPL Version 2 or to extend the choice of license to its licensees as provided above. However, if you add GPL Version 2 code and therefore, elected the GPL Version 2 license, then the option applies only if the new code is made subject to such option by the copyright holder.

/*
 * DO NOT ALTER OR REMOVE COPYRIGHT NOTICES OR THIS HEADER.
 *
 * Copyright (c) YYYY Oracle and/or its affiliates. All rights reserved.
 *
 * The contents of this file are subject to the terms of either the GNU
 * General Public License Version 2 only ("GPL") or the Common Development
 * and Distribution License("CDDL") (collectively, the "License"). You
 * may not use this file except in compliance with the License. You can
 * obtain a copy of the License at
 * http://glassfish.java.net/public/CDDL+GPL_1_1.html
 * or packager/legal/LICENSE.txt. See the License for the specific
 * language governing permissions and limitations under the License.
 *
 * When distributing the software, include this License Header Notice in each
 * file and include the License file at packager/legal/LICENSE.txt.
* GPL Classpath Exception:
* Oracle designates this particular file as subject to the "Classpath"
* exception as provided by Oracle in the GPL Version 2 section of the License
* file that accompanied this code.
*
* Modifications:
* If applicable, add the following below the License Header, with the fields
* enclosed by brackets [] replaced by your own identifying information:
* "Portions Copyright [year] [name of copyright owner]"
*
* Contributor(s):
* If you wish your version of this file to be governed by only the CDDL or
* only the GPL Version 2, indicate your decision by adding "[Contributor]
* elects to include this software in this distribution under the [CDDL or GPL
* Version 2] license." If you don't indicate a single choice of license, a
* recipient has the option to distribute your version of this file under
* either the CDDL, the GPL Version 2 or to extend the choice of license to
* its licensees as provided above. However, if you add GPL Version 2 code
* and therefore, elected the GPL Version 2 license, then the option applies
* only if the new code is made subject to such option by the copyright
* holder.
*/

COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)Version 1.1

1. Definitions.

1.1. "Contributor" means each individual or entity that creates or contributes to the creation of Modifications.

1.2. "Contributor Version" means the combination of the Original Software, prior Modifications used by a
Contributor (if any), and the Modifications made by that particular Contributor.

1.3. "Covered Software" means (a) the Original Software, or (b) Modifications, or (c) the combination of files
containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. "Executable" means the Covered Software in any form other than Source Code.

1.5. "Initial Developer" means the individual or entity that first makes Original Software available under this
License.

1.6. "Larger Work" means a work which combines Covered Software or portions thereof with code not governed
by the terms of this License.

1.7. "License" means this document.

1.8. "Licensable" means having the right to grant, to the maximum extent possible, whether at the time of the
initial grant or subsequently acquired, any and all of the rights conveyed herein.
1.9. "Modifications" means the Source Code and Executable form of any of the following:

A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

B. Any new file that contains any part of the Original Software or previous Modification; or

C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. "Original Software" means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. "Patent Claims" means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. "Source Code" means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. "You" (or "Your") means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, "You" includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims,
each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.
You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients' rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.


You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.

Oracle is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it
5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABILITY, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as "Participant") alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. If You assert a patent infringement claim against Participant alleging that the Participant Software directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

6.4. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN
INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a "commercial item," as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of "commercial computer software" (as that term is defined at 48 C.F.R. ? 252.227-7014(a)(1)) and "commercial computer software documentation" as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

--------

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)
The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.
The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc. 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301, USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software—-to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION
0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification"). Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.
In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the
conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF
ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.

Copyright (C)

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301, USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be
mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

"CLASSPATH" EXCEPTION TO THE GPL VERSION 2

Certain source files distributed by Oracle are subject to the following clarification and special exception to the GPL Version 2, but only where Oracle has expressly included in the particular source file's header the words "Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the License file that accompanied this code."

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License Version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

DO NOT TRANSLATE OR LOCALIZE.

%% The following software may be included in this product: ASM
Use of any of this software is governed by the terms of the license below:

Copyright (c) 2000-2005 INRIA, France Telecom
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of the copyright holders nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

%% The following software may be included in this product: Jettison
Use of any of this software is governed by the terms of the license below:

Copyright 2006 Envoi Solutions LLC

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

-----------------------------------------------------------------------------

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION
1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to
You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

1. You must give any other recipients of the Work or Derivative Works a copy of this License; and

2. You must cause any modified files to carry prominent notices stating that You changed the files; and

3. You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

4. If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or
conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "["] replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner] Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at http://www.apache.org/licenses/LICENSE-2.0 Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

%% The following software may be included in this product: Jackson

Use of any of this software is governed by the terms of the license below:

Jackson is dual-licensed under two alternative popular Open Source licenses: Apache (AL 2.0) and Gnu Lesser GPL (LGPL 2.1). You choose one or the other, as necessary (if you want to redistribute the code for use, you do not need license), and abide by the license rules as defined by the respective license agreement (and only that one).
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce,
prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

1. You must give any other recipients of the Work or Derivative Works a copy of this License; and

2. You must cause any modified files to carry prominent notices stating that You changed the files; and

3. You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

4. If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
APPENDIX: How to apply the Apache License to your work

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "["] replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner] Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at http://www.apache.org/licenses/LICENSE-2.0 Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* 
*   https://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/
/**
* Handles an I/O event or intercepts an I/O operation, and forwards it to its next handler in
* its [@link ChannelPipeline].
*
* <h3>Sub-types</h3>
* <p>
* [@link ChannelHandler] itself does not provide many methods, but you usually have to implement one of its
* subtypes:
* <ul>
* <li>[@link ChannelInboundHandler] to handle inbound I/O events, and</li>
* <li>[@link ChannelOutboundHandler] to handle outbound I/O operations.</li>
* </ul>
* </p>
* <p>
* Alternatively, the following adapter classes are provided for your convenience:
* <ul>
* <li>[@link ChannelInboundHandlerAdapter] to handle inbound I/O events,</li>
* <li>[@link ChannelOutboundHandlerAdapter] to handle outbound I/O operations, and</li>
* <li>[@link ChannelDuplexHandler] to handle both inbound and outbound events</li>
* </ul>
* </p>
* <p>
* For more information, please refer to the documentation of each subtype.
* </p>
* 
* <h3>The context object</h3>
* <p>
* A [@link ChannelHandler] is provided with a [@link ChannelHandlerContext]
* object. A [@link ChannelHandlerContext] is supposed to interact with the
* [@link ChannelPipeline] it belongs to via a context object. Using the
* context object, the [@link ChannelHandlerContext] can pass events upstream or
* downstream, modify the pipeline dynamically, or store the information
* (using [@link AttributeKey]s) which is specific to the handler.
* 
* <h3>State management</h3>
* 
* A [@link ChannelHandler] often needs to store some stateful information.
* The simplest and recommended approach is to use member variables:

```java
* public interface Message {
*   // your methods here
* }
*
* public class DataServerHandler extends @link SimpleChannelInboundHandler<Message> {
*
*     private boolean loggedIn;
*
*     @Override
*     public void channelRead0(@link ChannelHandlerContext ctx, Message message) {
*         if (message instanceof LoginMessage) {
*             authenticate((LoginMessage) message);
*             loggedIn = true;
*         } else (message instanceof GetDataMessage) {
*             if (loggedIn) {
*                 ctx.writeAndFlush(fetchSecret((GetDataMessage) message));
*             } else {
*                 fail();
*             }
*         }
*     }
* }
*
* // Create a new handler instance per channel.
* // See @link ChannelInitializer#initChannel(Channel).
* public class DataServerInitializer extends @link ChannelInitializer<Channel> {
*     @Override
*     public void initChannel(@link Channel channel) {
*         channel.pipeline().addLast("handler", new DataServerHandler());
*     }
* }
* </pre>
```

* Because the handler instance has a state variable which is dedicated to
* one connection, you have to create a new handler instance for each new
* channel to avoid a race condition where a unauthenticated client can get
* the confidential information:

```java
* // Create a new handler instance per channel.
* // See @link ChannelInitializer#initChannel(Channel).
* public class DataServerInitializer extends @link ChannelInitializer<Channel> {
*     @Override
*     public void initChannel(@link Channel channel) {
*         channel.pipeline().addLast("handler", new DataServerHandler());
*     }
* }
* </pre>
```

* Although it's recommended to use member variables to store the state of a
* handler, for some reason you might not want to create many handler instances.
* In such a case, you can use @link AttributeKey)s which is provided by
* @link ChannelHandlerContext):

---

Open Source Used In security_automation identity-guest-cs 1050
* <pre>
* public interface Message {
*     // your methods here
* }
*
* {@code @Sharable}
* public class DataServerHandler extends {@link SimpleChannelInboundHandler}&lt;Message&gt; { [91x776]*
*     private final {@link AttributeKey}&lt;{@link Boolean}&gt; auth =
*         {@link AttributeKey#valueOf(String) AttributeKey.valueOf("auth")};
*
*     [{@code @Override}
*     public void channelRead({@link ChannelHandlerContext} ctx, Message message) {
*         {@link Attribute}&lt;{@link Boolean}&gt; attr = ctx.attr(auth);
*         if (message instanceof LoginMessage) {
*             authenticate((LoginMessage) o);
*             attr.set(true);
*         } else (message instanceof GetDataMessage) {
*             if (Boolean.TRUE.equals(attr.get())) {
*                 ctx.writeAndFlush(fetchSecret((GetDataMessage) o));
*             } else {
*                 fail();
*             }
*         }
*     }
* }
* ...
* }
* </pre>

Now that the state of the handler is attached to the {@link ChannelHandlerContext}, you can add the
same handler instance to different pipelines:

* <pre>
* public class DataServerInitializer extends {@link ChannelInitializer}&lt;{@link Channel}&gt; { [91x341]*
*     private static final DataServerHandler <b>SHARED</b> = new DataServerHandler();
* 
*     [{@code @Override}
*     public void initChannel({@link Channel} channel) {
*         channel.pipeline().addLast("handler", <b>SHARED</b>);
*     }
* }
* </pre>

* <h4>The {@code @Sharable} annotation</h4>

* In the example above which used an {@link AttributeKey},
* you might have noticed the {@code @Sharable} annotation.
* <p>
* If a {@link ChannelHandler} is annotated with the {@code @Sharable}
* annotation, it means you can create an instance of the handler just once and
* add it to one or more [@link ChannelPipeline]s multiple times without
* a race condition.
* <p>
* If this annotation is not specified, you have to create a new handler
* instance every time you add it to a pipeline because it has unshared state
* such as member variables.
* <p>
* This annotation is provided for documentation purpose, just like
* <h3>Additional resources worth reading</h3>
* <p>
* Please refer to the [@link ChannelHandler], and
* [@link ChannelPipeline] to find out more about inbound and outbound operations,
* what fundamental differences they have, how they flow in a pipeline, and how to handle
* the operation in your application.
*/

Found in path(s):
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/ChannelHandler.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2020 The Netty Project
* 
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* 
*  https://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/StacklessClosedChannelException.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/socket/DuplexChannelConfig.java

No license file was found, but licenses were detected in source scan.
* Copyright 2014 The Netty Project
* 
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* 
* https://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/ReflectiveChannelFactory.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/ChannelFactory.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/embedded/EmbeddedChannelId.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/PendingWriteQueue.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/AbstractEventLoop.java
No license file was found, but licenses were detected in source scan.

~ Copyright 2012 The Netty Project
~ 
~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~ 
~ https://www.apache.org/licenses/LICENSE
2.0
~ 
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.

Found in path(s):
No license file was found, but licenses were detected in source scan.
/*
 * Copyright 2013 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/oio/OioByteStreamChannel.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2016 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/WriteBufferWaterMark.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/PreferHeapByteBufAllocator.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/SelectStrategy.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/bootstrap/BootstrapConfig.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2013 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 *   https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/ChannelOutboundInvoker.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/ChannelInboundInvoker.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/DefaultSelectStrategy.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/bootstrap/AbstractBootstrapConfig.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/socket/DuplexChannel.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/SelectStrategyFactory.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/group/VoidChannelGroupFuture.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/group/ChannelGroupException.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/ChannelProgressivePromise.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/ChannelProgressivePromise.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/ChannelProgressivePromise.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/DefaultChannelProgressivePromise.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/DefaultMessageSizeEstimator.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/DefaultMessageSizeEstimator.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/DefaultMessageSizeEstimator.java

Open Source Used In security_automation identity-guest-cs 1055
jar/io/netty/channel/DefaultAddressedEnvelope.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/DefaultChannelId.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/io/SelectedSelectionKeySet.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/socket/oio/OioServerSocketChannelConfig.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/ChannelOutboundBuffer.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/ConnectTimeoutException.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/MessageSizeEstimator.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/ChannelAdapter.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/group/ChannelGroup.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/ChannelProgressiveFutureListener.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/socket/oio/OioSocketChannelConfig.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/Channel.inboundHandler.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/group/ChannelMatcher.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/group/ChannelMatchers.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/group/DefaultChannelGroup.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/AbstractEventLoopGroup.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/AddressedEnvelope.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/bootstrap/ChannelFactory.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2017 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 */
* https://www.apache.org/licenses/LICENSE-2.0

* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-
jar/io/netty/channel/socket/oio/DefaultOioDatagramChannelConfig.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-
jar/io/netty/bootstrap/FailedChannel.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-
jar/io/netty/channel/PendingBytesTracker.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-
jar/io/netty/channel/socket/ChannelOutputShutdownException.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-
jar/io/netty/channel/internal/ChannelUtils.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-
jar/io/netty/channel/socket/ChannelOutputShutdownEvent.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-
jar/io/netty/channel/internal/package-info.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-
jar/io/netty/channel/socket/ChannelInputShutdownReadComplete.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-
jar/io/netty/channel/socket/oio/OioDatagramChannelConfig.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-
jar/io/netty/channel/DelegatingChannelPromiseNotifier.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2019 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
Found in path(s):
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/ExtendedClosedChannelException.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/EventLoopTaskQueueFactory.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/ChannelHandlerMask.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2015 The Netty Project
* 
* The Netty Project licenses this file to you under the Apache License, version 2.0 (the
* "License"); you may not use this file except in compliance with the License. You may obtain a
* copy of the License at:
* 
* https://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/CoalescingBufferQueue.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2017 The Netty Project
* 
* The Netty Project licenses this file to you under the Apache License, version 2.0 (the
* "License"); you may not use this file except in compliance with the License. You may obtain a
* copy of the License at:
* 
* https://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/
/*
 * Copyright 2012 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 *  https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

/*
 * Copyright 2012 The Netty Project
 * 
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * 
 *  https://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/VoidChannelPromise.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/package-info.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/socket/io/OioServerSocketChannel.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/socket/oio/OioServerSocketChannel.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/DefaultFileRegion.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/socket/nio/ProtocolFamilyConverter.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/socket/nio/NioEventLoopGroup.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/nio/NioEventLoopGroup.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/DefaultChannelPromise.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/DefaultChannelGroupFuture.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/bootstrap/package-info.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/socket/SocketChannel.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/AbstractChannelHandlerContext.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/ThreadPerChannelEventLoop.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/AbstractOioMessageChannel.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/DefaultChannelConfig.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/ServerChannel.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/oio/OioEventLoopGroup.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/socket/DefaultDatagramChannelConfig.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/socket/DefaultDatagramChannel.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/ChannelInboundHandler.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/ChannelOption.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/AdaptiveRecvByteBufAllocator.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/socket/nio/NioDatagramChannel.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/group/ChannelGroupFutureListener.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/AbstractServerChannel.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/FileRegion.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/ChannelGroupFutureListener.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/socket/InternetProtocolFamily.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/socket/nio/NioDatagramChannelConfig.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/socket/DatagramChannelConfig.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/nio/package-info.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/ChannelPipeline.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/DefaultEventLoopGroup.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/ChannelOutboundHandler.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/AbstractChannel.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/bootstrap/ServerBootstrap.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/ChannelOutboundHandlerAdapter.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/ChannelInboundHandlerAdapter.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/group/package-info.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/ChannelDuplexHandler.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/ChannelException.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2018 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You must obtain a copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */
Found in path(s):
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/socket/nio/NioChannelOption.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/SimpleUserEventChannelHandler.java
No license file was found, but licenses were detected in source scan.

*/

* Copyright 2015 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/pool/ChannelPoolHandler.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/pool/ChannelPool.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/pool/AbstractChannelPoolHandler.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/pool/AbstractChannelPoolMap.java
* /opt/ cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/pool/ChannelHealthChecker.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/pool/DefaultMaxBytesRecvByteBufAllocator.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/pool/MaxBytesRecvByteBufAllocator.java
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-jar/io/netty/channel/pool/MaxMessagesRecvByteBufAllocator.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 *   https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-
  jar/io/netty/channel/pool/ServerChannelRecvByteBufAllocator.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2021 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 *   https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1265673550_1643922077.24/0/netty-transport-4-1-69-final-sources-
  jar/io/netty/channel/ServerChannelRecvByteBufAllocator.java
No license file was found, but licenses were detected in source scan.
# The Netty Project licenses this file to you under the Apache License,
# version 2.0 (the "License"); you may not use this file except in compliance
# with the License. You may obtain a copy of the License at:
# distributed under the License is distributed on an "AS IS" BASIS, WITHOUT

Found in path(s):

1.173 netty-resolver-dns-native-macos
4.1.69.Final
1.173.1 Available under license :
No license file was found, but licenses were detected in source scan.

~ Copyright 2019 The Netty Project
~
~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~
~ https://www.apache.org/licenses/LICENSE-2.0
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.

Found in path(s):
No license file was found, but licenses were detected in source scan.

/ *
* Copyright 2019 The Netty Project
* *
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* *
* https://www.apache.org/licenses/LICENSE-2.0
* *
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
1.174 gson 2.8.8

1.174.1 Available under license:

No license file was found, but licenses were detected in source scan.

2018 Immutables Authors and Contributors

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Found in path(s):
* /opt/ cola/ permits/ 1221995412_1635506962.82/0/gson-2-8-8-sources-1-jar/META-INF/ maven/ org.immutables/ gson/pom.xml
No license file was found, but licenses were detected in source scan.

/*
Copyright 2015 Immutables Authors and Contributors

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0
Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
*/

Found in path(s):
- /opt/cola/permits/1221995412_1635506962.82/0/gson-2-8-8-sources-1.jar/org/immutables/gson/stream/JsonParserReader.java
- /opt/cola/permits/1221995412_1635506962.82/0/gson-2-8-8-sources-1.jar/org/immutables/gson/adapter/ExpectedSubtypesAdapter.java
- /opt/cola/permits/1221995412_1635506962.82/0/gson-2-8-8-sources-1.jar/org/immutables/gson/stream/GsonMessageBodyProvider.java
- /opt/cola/permits/1221995412_1635506962.82/0/gson-2-8-8-sources-1.jar/org/immutables/gson/stream/JsonGeneratorWriter.java
- /opt/cola/permits/1221995412_1635506962.82/0/gson-2-8-8-sources-1.jar/org/immutables/gson/Gson.java

1.175 rxjava 1.3.8
1.175.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 *
 * Original License: https://github.com/JCTools/JCTools/blob/master/LICENSE
 */

Found in path(s):
- /opt/cola/permits/1199767188_1630583074.7/0/rxjava-1-3-8-sources-4.jar/rx/internal/unsafe/Pow2.java

No license file was found, but licenses were detected in source scan.

/*
 * Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*
* Original License: https://github.com/JCTools/JCTools/blob/master/LICENSE
* Original location: https://github.com/JCTools/JCTools/blob/master/jctools-
core/src/main/java/org/jctools/queues/atomic/BaseLinkedAtomicQueue.java
*/

Found in path(s):
* /opt/cola/permits/1199767188_1630583074.7/0/rxjava-1-3-8-sources-4-
  jar/rx/internal/util/atomic/BaseLinkedAtomicQueue.java

No license file was found, but licenses were detected in source scan.

/**
* Copyright 2016 Netflix, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not
* use this file except in compliance with the License. You may obtain a copy of
* the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1199767188_1630583074.7/0/rxjava-1-3-8-sources-4-
  jar/rx/internal/schedulers/SchedulerLifecycle.java
* /opt/cola/permits/1199767188_1630583074.7/0/rxjava-1-3-8-sources-4-
  jar/rx/internal/operators/OnSubscribeReduce.java
* /opt/cola/permits/1199767188_1630583074.7/0/rxjava-1-3-8-sources-4-
  jar/rx/internal/util/SuppressAnimalSniffer.java
* /opt/cola/permits/1199767188_1630583074.7/0/rxjava-1-3-8-sources-4-
  jar/rx/internal/operators/OnSubscribeTakeLastOne.java
* /opt/cola/permits/1199767188_1630583074.7/0/rxjava-1-3-8-sources-4-
  jar/rx/internal/operators/DeferredScalarSubscriber.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2010 The Guava Authors
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 * Originally from https://code.google.com/p/guava-libraries/source/browse/guava/src/com/google/common/annotations/Beta.java
 */

Found in path(s):
/*
 * Copyright 2018 Netflix, Inc.
 * Licensed under the Apache License, Version 2.0 (the "License");
 */

No license file was found, but licenses were detected in source scan.
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1199767188_1630583074.7/0/rxjava-1-3-8-sources-4-jar/rx/internal/operators/OperatorGroupByEvicting.java
No license file was found, but licenses were detected in source scan.

====

Copyright 2014 Netflix, Inc.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
====

This package "rx.internal.operators" is for internal implementation details and can change at any time.

It is excluded from the public Javadocs (http://netflix.github.io/RxJava/javadoc/) and should not be relied upon by any code.

In short, changes to public signatures of these classes will not be accounted for in the versioning of RxJava.

Found in path(s):
* /opt/cola/permits/1199767188_1630583074.7/0/rxjava-1-3-8-sources-4-jar/rx/internal/operators/README.md
No license file was found, but licenses were detected in source scan.

/*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0

* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.

* Original License: https://github.com/JCTools/JCTools/blob/master/LICENSE
*/

Found in path(s):
* /opt/cola/permits/1199767188_1630583074.7/0/rxjava-1-3-8-sources-4-jar/rx/internal/util/atomic/MpscLinkedAtomicQueue.java

No license file was found, but licenses were detected in source scan.

/**
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*
* Original License: https://github.com/JCTools/JCTools/blob/master/LICENSE
* Original location: https://github.com/JCTools/JCTools/blob/master/jctools-queues/SpmcArrayQueue.java
*/

Found in path(s):
* /opt/cola/permits/1199767188_1630583074.7/0/rxjava-1-3-8-sources-4-jar/rx/internal/util/unsafe/SpmcArrayQueue.java

No license file was found, but licenses were detected in source scan.

/**
* Copyright 2017 Netflix, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1199767188_1630583074.7/0/rxjava-1-3-8-sources-4.jar/rx/internal/operators/OnSubscribeFlatMapSingle.java
* /opt/cola/permits/1199767188_1630583074.7/0/rxjava-1-3-8-sources-4.jar/rx/internal/operators/OnSubscribeFlatMapCompletable.java

No license file was found, but licenses were detected in source scan.

/*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
* Original License: https://github.com/JCTools/JCTools/blob/master/LICENSE
*/

Found in path(s):
* /opt/cola/permits/1199767188_1630583074.7/0/rxjava-1-3-8-sources-4.jar/rx/internal/unsafe/MpmcArrayQueue.java

No license file was found, but licenses were detected in source scan.

/*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.

* Original License: https://github.com/JCTools/JCTools/blob/master/LICENSE
*/

Found in path(s):
* /opt/cola/permits/1199767188_1630583074.7/0/rxjava-1-3-8-sources-4-jar/rx/internal/util/atomic/SpscUnboundedAtomicArrayQueue.java

No license file was found, but licenses were detected in source scan.

/**
 * Copyright 2014 Netflix, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1199767188_1630583074.7/0/rxjava-1-3-8-sources-4-jar/rx/internal/operators/CompletableOnSubscribeConcat.java
* /opt/cola/permits/1199767188_1630583074.7/0/rxjava-1-3-8-sources-4-jar/rx/internal/operators/SingleOperatorCast.java
* /opt/cola/permits/1199767188_1630583074.7/0/rxjava-1-3-8-sources-4-jar/rx/observables/ConnectableObservable.java
* /opt/cola/permits/1199767188_1630583074.7/0/rxjava-1-3-8-sources-4-jar/rx/internal/operators/CompletableOnSubscribeMergeArray.java
* /opt/cola/permits/1199767188_1630583074.7/0/rxjava-1-3-8-sources-4-jar/rx/functions/Func7.java
* /opt/cola/permits/1199767188_1630583074.7/0/rxjava-1-3-8-sources-4-jar/rx/plugins/RxJavaPlugins.java
* /opt/cola/permits/1199767188_1630583074.7/0/rxjava-1-3-8-sources-4-jar/rx/subscriptions/BooleanSubscription.java
* /opt/cola/permits/1199767188_1630583074.7/0/rxjava-1-3-8-sources-4-jar/rx/exceptions/MissingBackpressureException.java
* /opt/cola/permits/1199767188_1630583074.7/0/rxjava-1-3-8-sources-4-jar/rx/internal/operators/OperatorSkipWhile.java
* /opt/cola/permits/1199767188_1630583074.70/rxjava-1-3-8-sources-4-jar/rx/external/operators/OperatorSingle.java
* /opt/cola/permits/1199767188_1630583074.70/rxjava-1-3-8-sources-4-jar/rx/exceptions/OnErrorNotImplementedException.java
* /opt/cola/permits/1199767188_1630583074.70/rxjava-1-3-8-sources-4-jar/rx/schedulers/ImmediateScheduler.java
* /opt/cola/permits/1199767188_1630583074.70/rxjava-1-3-8-sources-4-jar/rx/external/operators/OperatorZipIterable.java
* /opt/cola/permits/1199767188_1630583074.70/rxjava-1-3-8-sources-4-jar/rx/subscriptions/RefCountSubscription.java
* /opt/cola/permits/1199767188_1630583074.70/rxjava-1-3-8-sources-4-jar/rx/functions/Func3.java
* /opt/cola/permits/1199767188_1630583074.70/rxjava-1-3-8-sources-4-jar/rx/internal/operators/CompletableOnSubscribeMerge.java
* /opt/cola/permits/1199767188_1630583074.70/rxjava-1-3-8-sources-4-jar/rx/observables/package-info.java
* /opt/cola/permits/1199767188_1630583074.70/rxjava-1-3-8-sources-4-jar/rx/subjects/BehaviorSubject.java
* /opt/cola/permits/1199767188_1630583074.70/rxjava-1-3-8-sources-4-jar/rx/external/operators/OperatorDelay.java
No license file was found, but licenses were detected in source scan.

/**
 * Copyright 2015 Netflix, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not
 * use this file except in compliance with the License. You may obtain a copy of
 * the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
*/opt/cola/permits/1199767188_1630583074.7/0/rxjava-1-3-8-sources-4-jar/rx/internal/producers/QueuedValueProducer.java
*/opt/cola/permits/1199767188_1630583074.7/0/rxjava-1-3-8-sources-4-jar/rx/internal/operators/OperatorTake.java
No license file was found, but licenses were detected in source scan.

/*
 *Licensed under the Apache License, Version 2.0 (the "License");
 *you may not use this file except in compliance with the License.
 *You may obtain a copy of the License at
 *
 *http://www.apache.org/licenses/LICENSE-2.0
 *
 *Unless required by applicable law or agreed to in writing, software
 *distributed under the License is distributed on an "AS IS" BASIS,
 *WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 *See the License for the specific language governing permissions and
 *limitations under the License.
 *
 *Original License: https://github.com/JCTools/JCTools/blob/master/LICENSE
 */

Found in path(s):
/ * /opt/cola/permits/1199767188_1630583074.7/0/rxjava-1-3-8-sources-4-jar/rx/internal/util/atomic/LinkedQueueNode.java
No license file was found, but licenses were detected in source scan.

/ **
 * Copyright 2015 Netflix, Inc.
 *
 *Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in
 *compliance with the License. You may obtain a copy of the License at
 *
 *http://www.apache.org/licenses/LICENSE-2.0
 *
 *Unless required by applicable law or agreed to in writing, software distributed under the License is
 *distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See
 *the License for the specific language governing permissions and limitations under the License.
Open Source Used In security_automation identity-guest-cs 1082

*/

Found in path(s):
* /opt/cola/permits/1199767188_1630583074.70/rxjava-1-3-8-sources-4-jar/rx/Single.java
* /opt/cola/permits/1199767188_1630583074.70/rxjava-1-3-8-sources-4-jar/rx/internal/operators/OnSubscribeCombineLatest.java
No license file was found, but licenses were detected in source scan.

/**
 * Copyright 2014 Netflix, Inc.
 * 
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 * http://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 * 
 */

Found in path(s):
* /opt/cola/permits/1199767188_1630583074.70/rxjava-1-3-8-sources-4-jar/rx/internal/util/OpenHashSet.java
No license file was found, but licenses were detected in source scan.

/**
 * Copyright 2015 Netflix, Inc.
 * 
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 * http://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1199767188_1630583074.70/rxjava-1-3-8-sources-4-jar/
This package contains code that relies on sun.misc.Unsafe. Before using it you MUST assert UnsafeAccess.isUnsafeAvailable() == true

Much of the code in this package comes from or is inspired by the JCTools project:
https://github.com/JCTools/JCTools

RxJava has a "zero dependency" policy for the core library, so if we do add it as a dependency, it won't be an externally visible dependency that results in a separate jar.

The license for the JCTools code is https://github.com/JCTools/JCTools/blob/master/LICENSE

As of June 10 2014 when this code was copied the LICENSE read as:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.
You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify, 
defend, and hold each Contributor harmless for any liability 
incurred by, or claims asserted against, such Contributor by reason 
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following 
boilerplate notice, with the fields enclosed by brackets "{}" 
replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate 
comment syntax for the file format. We also recommend that a 
file or class name and description of purpose be included on the 
same "printed page" as the copyright notice for easier 
identification within third-party archives.

Copyright {yyyy} [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); 
you may not use this file except in compliance with the License. 
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software 
distributed under the License is distributed on an "AS IS" BASIS, 
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. 
See the License for the specific language governing permissions and 
limitations under the License.

Found in path(s):
* /opt/cola/permits/1199767188_1630583074.7/0/rxjava-1-3-8-sources-4.jar/rx/internal/util/unsafe/README.md
No license file was found, but licenses were detected in source scan.

/ *
* Copyright 2016 Netflix, Inc.
* *
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* *
* http://www.apache.org/licenses/LICENSE-2.0
* *
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1199767188_1630583074.70/rxjava-1-3-8-sources-4-jar/rx/CompletableEmitter.java
* /opt/cola/permits/1199767188_1630583074.70/rxjava-1-3-8-sources-4-jar/rx/CompletableSubscriber.java
* /opt/cola/permits/1199767188_1630583074.70/rxjava-1-3-8-sources-4-jar/rx/SingleEmitter.java
* /opt/cola/permits/1199767188_1630583074.70/rxjava-1-3-8-sources-4-jar/rx/internal/operators/CompletableFromEmitter.java

No license file was found, but licenses were detected in source scan.

/**
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*
* Original License: https://github.com/JCTools/JCTools/blob/master/LICENSE
*/

Found in path(s):
* /opt/cola/permits/1199767188_1630583074.70/rxjava-1-3-8-sources-4-jar/rx/internal/util/unsafe/BaseLinkedQueue.java

No license file was found, but licenses were detected in source scan.

/**
* Copyright 2014 Netflix, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and

---

Open Source Used In security_automation identity-guest-cs 1088
* limitations under the License.
*/
/**
 * This mechanism for getting UNSAFE originally from:
 *
 * Original License: https://github.com/JCTools/JCTools/blob/master/LICENSE
 */

Found in path(s):
* /opt/cola/permits/1199767188_1630583074.7/0/rxjava-1-3-8-sources-4-jar/rx/internal/util/unsafe/UnsafeAccess.java

No license file was found, but licenses were detected in source scan.

/**
 * Copyright 2016 Netflix, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1199767188_1630583074.7/0/rxjava-1-3-8-sources-4-jar/rx/internal/operators/OnSubscribeDetach.java
* /opt/cola/permits/1199767188_1630583074.7/0/rxjava-1-3-8-sources-4-jar/rx/Emitter.java
* /opt/cola/permits/1199767188_1630583074.7/0/rxjava-1-3-8-sources-4-jar/rx/internal/operators/SingleTakeUntilObservable.java
* /opt/cola/permits/1199767188_1630583074.7/0/rxjava-1-3-8-sources-4-jar/rx/observables/package-info.java
* /opt/cola/permits/1199767188_1630583074.7/0/rxjava-1-3-8-sources-4-jar/rx/observers/SafeCompletableSubscriber.java
* /opt/cola/permits/1199767188_1630583074.7/0/rxjava-1-3-8-sources-4-jar/rx/internal/operators/OnSubscribeThrow.java
* /opt/cola/permits/1199767188_1630583074.7/0/rxjava-1-3-8-sources-4-jar/rx/internal/observable/OnSubscribeThrow.java
* /opt/cola/permits/1199767188_1630583074.7/0/rxjava-1-3-8-sources-4-jar/rx/observers/package-info.java
* /opt/cola/permits/1199767188_1630583074.7/0/rxjava-1-3-8-sources-4-jar/rx/internal/operators/CompletableFlatMapSingleToCompletable.java
* /opt/cola/permits/1199767188_1630583074.7/0/rxjava-1-3-8-sources-4-jar/rx/internal/operators/CompletableFlatMapSingleToCompletable.java
* /opt/cola/permits/1199767188_1630583074.7/0/rxjava-1-3-8-sources-4-jar/rx/observers/package-info.java
* /opt/cola/permits/1199767188_1630583074.7/0/rxjava-1-3-8-sources-4-jar/rx/internal/operators/CompletableFlatMapSingleToCompletable.java
No license file was found, but licenses were detected in source scan.

/*
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
* Original License: https://github.com/JCTools/JCTools/blob/master/LICENSE
*/

Found in path(s):
* /opt/cola/permits/1199767188_1630583074.7/0/rxjava-1-3-8-sources-4-jar/rx/internal/util/unsafe/ConcurrentCircularArrayQueue.java
No license file was found, but licenses were detected in source scan.

/**
 * Copyright one 2014 Netflix, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1199767188_1630583074.7/0/rxjava-1-3-8-sources-4-jar/rx/internal/operators/OnSubscribeToMultimap.java
No license file was found, but licenses were detected in source scan.

/**
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 *
 * Original License: https://github.com/JCTools/JCTools/blob/master/LICENSE
 */
No license file was found, but licenses were detected in source scan.

/*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*
* Original License: https://github.com/JCTools/JCTools/blob/master/LICENSE
* Original location: https://github.com/JCTools/JCTools/blob/master/jctools-
core/src/main/java/org/jctools/queues/SpscArrayQueue.java
*/

No license file was found, but licenses were detected in source scan.

/*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*
* Original License: https://github.com/JCTools/JCTools/blob/master/LICENSE
* Original location: https://github.com/JCTools/JCTools/blob/master/jctools-
core/src/main/java/org/jctools/queues/AtomicReferenceArrayQueue.java
*/

Found in path(s):
* /opt/cola/permits/1199767188_1630583074.7/0/rxjava-1-3-8-sources-4.jar/rx/internal/util/atomic/AtomicReferenceArrayQueue.java
No license file was found, but licenses were detected in source scan.

/**
 * Copyright 2014 Netflix, Inc.
 * <p/>
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * <p/>
 * http://www.apache.org/licenses/LICENSE-2.0
 * <p/>
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1199767188_1630583074.7/0/rxjava-1-3-8-sources-4.jar/rx/internal/util/ScalarSynchronousSingle.java
No license file was found, but licenses were detected in source scan.

/*
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * <p/>
 * http://www.apache.org/licenses/LICENSE-2.0
 * <p/>
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 *
 * Original License: https://github.com/JCTools/JCTools/blob/master/LICENSE
 */

Found in path(s):
* /opt/cola/permits/1199767188_1630583074.7/0/rxjava-1-3-8-sources-4.jar/rx/internal/util/atomic/SpscLinkedAtomicQueue.java
No license file was found, but licenses were detected in source scan.
/*
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 *
 * Original License: https://github.com/JCTools/JCTools/blob/master/LICENSE
 * Original location: https://github.com/JCTools/JCTools/blob/master/jctools-
 * core/src/main/java/org/jctools/queues/MpscLinkedQueue.java
 */

No license file was found, but licenses were detected in source scan.

Copyright 2014 Netflix, Inc.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

No license file was found, but licenses were detected in source scan.
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License is
distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
or implied. See
* the License for the specific language governing permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1199767188_1630583074.7/0/rxjava-1-3-8-sources-4-jar/rx/internal/util/UtilityFunctions.java
* /opt/cola/permits/1199767188_1630583074.7/0/rxjava-1-3-8-sources-4-jar/rx/Observable.java
No license file was found, but licenses were detected in source scan.

/**
 * Copyright 2014 Netflix, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
/**
 * This plugin class provides 2 ways to customize { @link Scheduler} functionality
 * 1. You may redefine entire schedulers, if you so choose. To do so, override
 * the 3 methods that return Scheduler (io(), computation(), newThread()).
 * 2. You may wrap/decorate an { @link Action0}, before it is handed off to a Scheduler. The system-
supplied Schedulers (Schedulers.ioScheduler, Schedulers.computationScheduler,
Scheduler.newThreadScheduler) all use this hook, so it's a convenient way to
modify Scheduler functionality without redefining Schedulers wholesale.
*
* Also, when redefining Schedulers, you are free to use/not use the onSchedule decoration hook.
* <p>
* See { @link RxJavaPlugins} or the RxJava GitHub Wiki for information on configuring plugins:
 */

Found in path(s):
* /opt/cola/permits/1199767188_1630583074.7/0/rxjava-1-3-8-sources-4-jar/rx/plugins/RxJavaSchedulersHook.java
No license file was found, but licenses were detected in source scan.
/**
 * Copyright 2014 Netflix, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not
 * use this file except in compliance with the License. You may obtain a copy of
 * the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/cola/permits/1199767188_1630583074.70/rxjava-1-3-8-sources-4-jar/rx/functions/Actions.java
* /opt/cola/permits/1199767188_1630583074.70/rxjava-1-3-8-sources-4-jar/rx/internal/operators/OperatorBufferWithTime.java
* /opt/cola/permits/1199767188_1630583074.70/rxjava-1-3-8-sources-4-jar/rx/internal/operators/OperatorMulticast.java
* /opt/cola/permits/1199767188_1630583074.70/rxjava-1-3-8-sources-4-jar/rx/internal/operators/OperatorMapPair.java
* /opt/cola/permits/1199767188_1630583074.70/rxjava-1-3-8-sources-4-jar/rx/internal/operators/OperatorEagerConcatMap.java
* /opt/cola/permits/1199767188_1630583074.70/rxjava-1-3-8-sources-4-jar/rx/internal/operators/OperatorDistinct.java
* /opt/cola/permits/1199767188_1630583074.70/rxjava-1-3-8-sources-4-jar/rx/internal/operators/ScheduledAction.java
* /opt/cola/permits/1199767188_1630583074.70/rxjava-1-3-8-sources-4-jar/rx/internal/operators/OperatorSampleWithObservable.java
* /opt/cola/permits/1199767188_1630583074.70/rxjava-1-3-8-sources-4-jar/rx/internal/operators/OnSubscribeTimerPeriodically.java
* /opt/cola/permits/1199767188_1630583074.70/rxjava-1-3-8-sources-4-jar/rx/internal/operators/ExecutorScheduler.java
* /opt/cola/permits/1199767188_1630583074.70/rxjava-1-3-8-sources-4-jar/rx/internal/operators/OperatorBufferWithSize.java
* /opt/cola/permits/1199767188_1630583074.70/rxjava-1-3-8-sources-4-jar/rx/internal/operators/OperatorDebounceWithSelector.java
* /opt/cola/permits/1199767188_1630583074.70/rxjava-1-3-8-sources-4-jar/rx/internal/operators/OperatorDebounceWithTime.java
* /opt/cola/permits/1199767188_1630583074.70/rxjava-1-3-8-sources-4-jar/rx/internal/operators/OperatorWindowWithStartEndObservable.java
* /opt/cola/permits/1199767188_1630583074.70/rxjava-1-3-8-sources-4-jar/rx/internal/util/RxThreadFactory.java
* /opt/cola/permits/1199767188_1630583074.70/rxjava-1-3-8-sources-4-jar/rx/internal/operators/OnSubscribeDelaySubscription.java

Found in path(s):
Open Source Used In security_automation identity-guest-cs 1098
No license file was found, but licenses were detected in source scan.

/*
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 *
 * Original License: https://github.com/JCTools/JCTools/blob/master/LICENSE
 */

Found in path(s):
* /opt/cola/permits/1199767188_1630583074.7/0/rxjava-1-3-8-sources-4-jar/rx/internal/util/unsafe/ConcurrentSequencedCircularArrayQueue.java

No license file was found, but licenses were detected in source scan.

/*
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 *
 * Original License: https://github.com/JCTools/JCTools/blob/master/LICENSE
 */

Found in path(s):
* /opt/cola/permits/1199767188_1630583074.7/0/rxjava-1-3-8-sources-4-jar/rx/internal/util/atomic/SpscExactAtomicArrayQueue.java
No license file was found, but licenses were detected in source scan.

/*
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 *
 * Original License: https://github.com/JCTools/JCTools/blob/master/LICENSE
 */

Found in path(s):
* /opt/cola/permits/1199767188_1630583074.70/rxjava-1-3-8-sources-4-jar/rx/annotations/Experimental.java

No license file was found, but licenses were detected in source scan.

/*
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 *
 * Original License: https://github.com/JCTools/JCTools/blob/master/LICENSE
 */

Found in path(s):
* /opt/cola/permits/1199767188_1630583074.70/rxjava-1-3-8-sources-4-jar/rx/annotations/Experimental.java

No license file was found, but licenses were detected in source scan.
/*
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 *
 * Original License: https://github.com/JCTools/JCTools/blob/master/LICENSE
 * Original location: https://github.com/JCTools/JCTools/blob/master/jctools-
 * core/src/main/java/org/jctools/queues/QueueProgressIndicators.java
 */

Found in path(s):
* /opt/cola/permits/1199767188_1630583074.70/rxjava-1-3-8-sources-4-
  jar/rx/internal/util/unsafe/QueueProgressIndicators.java

No license file was found, but licenses were detected in source scan.

/*
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 *
 * Original License: https://github.com/JCTools/JCTools/blob/master/LICENSE
 * Original location: https://github.com/JCTools/JCTools/blob/master/jctools-
 * core/src/main/java/org/jctools/queues/MessagePassingQueue.java
 */

Found in path(s):
* /opt/cola/permits/1199767188_1630583074.70/rxjava-1-3-8-sources-4-
  jar/rx/internal/util/unsafe/MessagePassingQueue.java
1.176 spring-jdbc 5.3.12

1.176.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2016 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 *   https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-jar/org/springframework/jdbc/core/PreparedStatementSetter.java
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-jar/org/springframework/jdbc/datasource/init/DatabasePopulator.java
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-jar/org/springframework/jdbc/datasource/lookup/DataSourceLookup.java
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-jar/org/springframework/jdbc/support/lob/LobHandler.java
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-jar/org/springframework/jdbc/core/RowCallbackHandler.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2015 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 *   https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-jar/org/springframework/jdbc/datasource/embedded/EmbeddedDatabase.java
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-jar/org/springframework/jdbc/datasource/init/ScriptStatementFailedException.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2008 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *      https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-jar/org/springframework/jdbc/support/xml/SqlXmlValue.java
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-jar/org/springframework/jdbc/support/incrementer/DB2MainframeSequenceMaxValueIncrementer.java
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-jar/org/springframework/jdbc/support/incrementer/DB2SequenceMaxValueIncrementer.java
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-jar/org/springframework/jdbc/core/DisposableSqlTypeValue.java
No license file was found, but licenses were detected in source scan.

/**
 * The classes in this package make JDBC easier to use and
 * reduce the likelihood of common errors. In particular, they:
 * <ul>
 * <li>Simplify error handling, avoiding the need for try/catch/finally
 * blocks in application code.
 * <li>Present exceptions to application code in a generic hierarchy of
 * unchecked exceptions, enabling applications to catch data access
 * exceptions without being dependent on JDBC, and to ignore fatal
 * exceptions there is no value in catching.
 * <li>Allow the implementation of error handling to be modified

Open Source Used In security_automation identity-guest-cs 1103
* to target different RDBMSes without introducing proprietary
* dependencies into application code.
* </ul>
* 
* <p>This package and related packages are discussed in Chapter 9 of
* a href="https://www.amazon.com/exec/obidos/tg/detail/-/0764543857/">Expert One-On-One J2EE Design and
Development</a>
* by Rod Johnson (Wrox, 2002).
*/

Found in path(s):
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-
jar/org/springframework/jdbc/package-info.java

No license file was found, but licenses were detected in source scan.

/*/ 
* Copyright 2002-2007 the original author or authors.
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
*      https://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-
jar/org/springframework/jdbc/support/incrementer/DataFieldMaxValueIncrementer.java

No license file was found, but licenses were detected in source scan.

/*/ 
* Copyright 2002-2020 the original author or authors.
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
*      https://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
  * limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-
  jar/org/springframework/jdbc/datasource/init/DatabasePopulatorUtils.java
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-
  jar/org/springframework/jdbc/datasource/SingleConnectionDataSource.java
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-
  jar/org/springframework/jdbc/core/namedparam/NamedParameterJdbcTemplate.java
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-
  jar/org/springframework/jdbc/core/metadata/CallParameterMetaData.java
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-
  jar/org/springframework/jdbc/core/metadata/CallMetaDataProviderFactory.java
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-
  jar/org/springframework/jdbc/core/support/JdbcUtils.java
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-
  jar/org/springframework/jdbc/support/SQLErrorCodesFactory.java
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-
  jar/org/springframework/jdbc/support/SQLExceptionTranslator.java
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-
  jar/org/springframework/jdbc/support/DatabaseStartupValidator.java
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-
  jar/org/springframework/jdbc/support/DatabaseMetaDataCallback.java
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-
  jar/org/springframework/jdbc/support/SQLErrorCodeSQLExceptionTranslator.java
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-
  jar/org/springframework/jdbc/datasource/LazyConnectionDataSourceProxy.java
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-
  jar/org/springframework/jdbc/datasource/DriverManagerDataSource.java
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-
  jar/org/springframework/jdbc/core/namedparam/ParsedSql.java
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-
  jar/org/springframework/jdbc/core/support/DatabaseMetaDataCallback.java
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-
  jar/org/springframework/jdbc/support/AbstractFallbackSQLExceptionTranslator.java
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-
  jar/org/springframework/jdbc/support/SqlParameterValue.java
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-
  jar/org/springframework/jdbc/support/DatabaseStartupValidator.java
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-
  jar/org/springframework/jdbc/datasource/LazyConnectionDataSourceProxy.java
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2012 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 */
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2002-2020 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
*   https://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
/**
 * Strategy interface for translating between [SQLException SQLExceptions]
 * and Spring's data access strategy-agnostic [DataAccessException]
 * hierarchy.
 *
 * Implementations can be generic (for example, using
 * [SQLException#getSQLState() SQLState] codes for JDBC) or wholly
 * proprietary (for example, using Oracle error codes) for greater precision.
 * 
 * @author Rod Johnson
 * @author Juergen Hoeller
 * @see org.springframework.dao.DataAccessException
 */

Found in path(s):
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-
jar/org/springframework/jdbc/support/SQLExceptionTranslator.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2017 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 *   https://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2013 the original author or authors.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 */
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-jar/org/springframework/jdbc/support/lob/AbstractLobHandler.java
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-jar/org/springframework/jdbc/support/incrementer/SqlServerMaxValueIncrementer.java
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-jar/org/springframework/jdbc/datasource/embedded/ConnectionProperties.java
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-jar/org/springframework/jdbc/support/incrementer/HsqlMaxValueIncrementer.java
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-jar/org/springframework/jdbc/support/rowset/SqlRowSetMetaData.java

No license file was found, but licenses were detected in source scan.

/*/ Copyright 2002-2014 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* *
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-jar/org/springframework/jdbc/support/incrementer/SqlServerMaxValueIncrementer.java
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-jar/org/springframework/jdbc/datasource/embedded/ConnectionProperties.java
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-jar/org/springframework/jdbc/support/incrementer/HsqlMaxValueIncrementer.java
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-jar/org/springframework/jdbc/support/rowset/SqlRowSetMetaData.java

No license file was found, but licenses were detected in source scan.

/*/ Copyright 2002-2021 the original author or authors
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* https://www.apache.org/licenses/LICENSE-2.0

* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-jar/org/springframework/jdbc/core/JdbcOperationsExtensions.kt

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2018 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *      https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

/**
 * Implement this interface when parameters need to be customized based
 * on the connection. We might need to do this to make use of proprietary
 * features, available only with a specific Connection type.
 *
 * @author Rod Johnson
 * @author Thomas Risberg
 * @see CallableStatementCreatorFactory#newCallableStatementCreator(ParameterMapper)
 * @see org.springframework.jdbc.object.StoredProcedure#execute(ParameterMapper)
 */

/**
 * Create a Map of input parameters, keyed by name.
 * @param con a JDBC connection. This is useful (and the purpose of this interface)
 * if we need to do something RDBMS-specific with a proprietary Connection
 * implementation class. This class conceals such proprietary details. However,
 * it is best to avoid using such proprietary RDBMS features if possible.
 * @return a Map of input parameters, keyed by name (never ( @code null))
 * @throws SQLException if an SQLException is encountered setting
 * parameter values (that is, there's no need to catch SQLException)
Found in path(s):
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-jar/org/springframework/jdbc/core/ParameterMapper.java
No license file was found, but licenses were detected in source scan.

Found in path(s):
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-jar/org/springframework/jdbc/core/namedparam/MapSqlParameterSourceExtensions.kt
No license file was found, but licenses were detected in source scan.

Found in path(s):
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-jar/org/springframework/jdbc/core/namedparam/NamedParameterUtils.java
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-
jar/org/springframework/jdbc/core/simple/AbstractJdbcCall.java
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-
jar/org/springframework/jdbc/support/incrementer/AbstractIdentityColumnMaxValueIncrementer.java
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-
jar/org/springframework/jdbc/core/metadata/CallMetaDataContext.java
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-
jar/org/springframework/jdbc/datasource/init/AbstractJdbcInsert.java
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-
jar/org/springframework/jdbc/datasource/init/SimpleJdbcInsert.java
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-
jar/org/springframework/jdbc/core/metadata/TableMetaDataContext.java
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-
jar/org/springframework/jdbc/datasource/init/UncategorizedScriptException.java
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-
jar/org/springframework/jdbc/datasource/init/CompositeDatabasePopulator.java
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-
jar/org/springframework/jdbc/datasource/init/CompositeDatabasePopulator.java
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-
jar/org/springframework/jdbc/core/metadata/GenericTableMetaDataProvider.java
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-
jar/org/springframework/jdbc/core/metadata/GenericTableMetaDataProviderFactory.java
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-
jar/org/springframework/jdbc/core/metadata/TableMetaDataProviderFactory.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2016 the original author or authors.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 */
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
/**
* Close this LobCreator session and free its temporarily created BLOBs and CLOBs.
* Will not need to do anything if using PreparedStatement's standard methods,
* but might be necessary to free database resources if using proprietary means.
* <p><b>NOTE</b>: Needs to be invoked after the involved PreparedStatements have
* been executed or the affected O/R mapping sessions have been flushed.
* Otherwise, the database resources for the temporary BLOBs might stay allocated.
*/

Found in path(s):
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-
  jar/org/springframework/jdbc/support/lob/LobCreator.java
No license file was found, but licenses were detected in source scan.

/ * Copyright 2002-2018 the original author or authors.
* *
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* *
* https://www.apache.org/licenses/LICENSE-2.0
* *
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-
  jar/org/springframework/jdbc/BadSqlGrammarException.java
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-
  jar/org/springframework/jdbc/core/CallableStatementCallback.java
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-
  jar/org/springframework/jdbc/object/SqlOperation.java
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-
  jar/org/springframework/jdbc/UncategorizedSQLException.java
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-
/*
 * Copyright 2002-2019 the original author or authors.
 * 
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 *      https://www.apache.org/licenses/LICENSE-2.0
 * 
 * No license file was found, but licenses were detected in source scan.
 */

Open Source Used in security_automation identity-guest-cs 1118
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-jar/org/springframework/jdbc/support/SQLStateSQLExceptionTranslator.java
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-jar/org/springframework/jdbc/core/support/AbstractLobStreamingResultSetExtractor.java
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-jar/org/springframework/jdbc/core/SqlParameterSource.java
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-jar/org/springframework/jdbc/core/MapSqlParameterSource.java
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-jar/org/springframework/jdbc/core/namedparam/AbstractSqlParameterSource.java
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-jar/org/springframework/jdbc/datasource/JdbcTransactionObjectSupport.java
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-jar/org/springframework/jdbc/support/r2dbc/SqlRowSet.java
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-jar/org/springframework/jdbcTemplate/BeanPropertySqlParameterSource.java
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-jar/org/springframework/jdbcTemplate/init/ResourceDatabasePopulator.java
* /opt/cola/permits/1221016716_1635514078.35/0/spring-jdbc-5-3-9-sources-jar/org/springframework/jdbcTemplate/object/SqlFunction.java

1.177 activation-api 1.2.0
1.177.1 Available under license:
COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.1

1. Definitions.

1.1. "Contributor" means each individual or entity that creates or contributes to the creation of Modifications.

1.2. "Contributor Version" means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. "Covered Software" means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original
Software with files containing Modifications, in each case including portions thereof.

1.4. "Executable" means the Covered Software in any form other than Source Code.

1.5. "Initial Developer" means the individual or entity that first makes Original Software available under this License.

1.6. "Larger Work" means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. "License" means this document.

1.8. "Licensable" means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. "Modifications" means the Source Code and Executable form of any of the following:

A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

B. Any new file that contains any part of the Original Software or previous Modification; or

C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. "Original Software" means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. "Patent Claims" means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. "Source Code" means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. "You" (or "Your") means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, "You" includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct
or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and
3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained
within the Covered Software, or any notices of licensing or any
descriptive text giving attribution to any Contributor or the
Initial Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in
Source Code form that alters or restricts the applicable version of
this License or the recipients’ rights hereunder. You may choose to
offer, and to charge a fee for, warranty, support, indemnity or
liability obligations to one or more recipients of Covered Software.
However, you may do so only on Your own behalf, and not on behalf of
the Initial Developer or any Contributor. You must make it
absolutely clear that any such warranty, support, indemnity or
liability obligation is offered by You alone, and You hereby agree
to indemnify the Initial Developer and every Contributor for any
liability incurred by the Initial Developer or such Contributor as a
result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under
the terms of this License or under the terms of a license of Your
choice, which may contain terms different from this License,
provided that You are in compliance with the terms of this License
and that the license for the Executable form does not attempt to
limit or alter the recipient’s rights in the Source Code form from
the rights set forth in this License. If You distribute the Covered
Software in Executable form under a different license, You must make
it absolutely clear that any terms which differ from this License
are offered by You alone, not by the Initial Developer or
Contributor. You hereby agree to indemnify the Initial Developer and
every Contributor for any liability incurred by the Initial
Developer or such Contributor as a result of any such terms You offer.


You may create a Larger Work by combining Covered Software with
other code not governed by the terms of this License and distribute
the Larger Work as a single product. In such a case, You must make
sure the requirements of this License are fulfilled for the Covered
Software.

4. Versions of the License.

4.1. New Versions.

Oracle is the initial license steward and may publish revised and/or
new versions of this License from time to time. Each version will be
given a distinguishing version number. Except as provided in Section
4.3, no one other than the license steward has the right to modify
this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the
Covered Software available under the terms of the version of the
License under which You originally received the Covered Software. If
the Initial Developer includes a notice in the Original Software
prohibiting it from being distributed or otherwise made available
under any subsequent version of the License, You must distribute and
make the Covered Software available under the terms of the version
of the License under which You originally received the Covered
Software. Otherwise, You may also choose to use, distribute or
otherwise make the Covered Software available under the terms of any
subsequent version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to create a new
license for Your Original Software, You may create and use a
modified version of this License if You: (a) rename the license and
remove any references to the name of the license steward (except to
note that the license differs from this License); and (b) otherwise
make it clear that the license contains terms which differ from this
License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS,
WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED,
INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE
IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR
NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF
THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE
DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY
OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING,
REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN
ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS
AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate
automatically if You fail to comply with terms herein and fail to
cure such breach within 30 days of becoming aware of the breach.
Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as "Participant") alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. If You assert a patent infringement claim against Participant alleging that the Participant Software directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

6.4. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH
LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a "commercial item," as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of "commercial computer software" (as that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and "commercial computer software documentation" as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys’ fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or
shall be deemed to constitute any admission of liability.

------------------------------------------------------------------------

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)

The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor
Boston, MA 02110-1335
USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation’s software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.
For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice
and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a
storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.
5. You are not required to accept this License, since you have not
signed it. However, nothing else grants you permission to modify or
distribute the Program or its derivative works. These actions are
prohibited by law if you do not accept this License. Therefore, by
modifying or distributing the Program (or any work based on the
Program), you indicate your acceptance of this License to do so, and all
its terms and conditions for copying, distributing or modifying the
Program or works based on it.

6. Each time you redistribute the Program (or any work based on the
Program), the recipient automatically receives a license from the
original licensor to copy, distribute or modify the Program subject to
these terms and conditions. You may not impose any further restrictions
on the recipients' exercise of the rights granted herein. You are not
responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent
infringement or for any other reason (not limited to patent issues),
conditions are imposed on you (whether by court order, agreement or
otherwise) that contradict the conditions of this License, they do not
excuse you from the conditions of this License. If you cannot distribute
so as to satisfy simultaneously your obligations under this License and
any other pertinent obligations, then as a consequence you may not
distribute the Program at all. For example, if a patent license would
not permit royalty-free redistribution of the Program by all those who
receive copies directly or indirectly through you, then the only way you
could satisfy both it and this License would be to refrain entirely from
distribution of the Program.

If any portion of this section is held invalid or unenforceable under
any particular circumstance, the balance of the section is intended to
apply and the section as a whole is intended to apply in other
circumstances.

It is not the purpose of this section to induce you to infringe any
patents or other property right claims or to contest validity of any
such claims; this section has the sole purpose of protecting the
integrity of the free software distribution system, which is implemented
by public license practices. Many people have made generous
contributions to the wide range of software distributed through that
system in reliance on consistent application of that system; it is up to
the author/donor to decide if he or she is willing to distribute
software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be
a consequence of the rest of this License.
8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF)
THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR
OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest
possible use to the public, the best way to achieve this is to make it
free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to
attach them to the start of each source file to most effectively convey
the exclusion of warranty; and each file should have at least the
"copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful, but
WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program; if not, write to the Free Software
Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1335 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this
when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type
`show w'. This is free software, and you are welcome to redistribute
it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the
appropriate parts of the General Public License. Of course, the commands
you use may be called something other than `show w' and `show c'; they
could even be mouse-clicks or menu items--whatever suits your program.
You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the program, if
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the
program `Gnomovision' (which makes passes at compilers) written by
James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program
into proprietary programs. If your program is a subroutine library, you
may consider it more useful to permit linking proprietary applications
with the library. If this is what you want to do, use the GNU Library
General Public License instead of this License.

#

Certain source files distributed by Oracle America, Inc. and/or its
affiliates are subject to the following clarification and special
exception to the GPLv2, based on the GNU Project exception for its
Classpath libraries, known as the GNU Classpath Exception, but only
where Oracle has expressly included in the particular source file's
header the words "Oracle designates this particular file as subject to
the "Classpath" exception as provided by Oracle in the LICENSE file
that accompanied this code."

You should also note that Oracle includes multiple, independent
programs in this software package. Some of those programs are provided
under licenses deemed incompatible with the GPLv2 by the Free Software
Foundation and others. For example, the package includes programs
licensed under the Apache License, Version 2.0. Such programs are
licensed to you under their original licenses.

Oracle facilitates your further distribution of this package by adding
the Classpath Exception to the necessary parts of its GPLv2 code, which
permits you to use that code in combination with other independent
modules not licensed under the GPLv2. However, note that this would
not permit you to commingle code under an incompatible license with
Oracle's GPLv2 licensed code by, for example, cutting and pasting such
code into a file also containing Oracle's GPLv2 licensed code and then
distributing the result. Additionally, if you were to remove the
Classpath Exception from any of the files to which it applies and
distribute the result, you would likely be required to license some or
all of the other code in that distribution under the GPLv2 as well, and
since the GPLv2 is incompatible with the license terms of some items
included in the distribution by Oracle, removing the Classpath
Exception could therefore effectively compromise your ability to further distribute the package.

Proceed with caution and we recommend that you obtain the advice of a lawyer skilled in open source matters before removing the Classpath Exception or making modifications to this package which may subsequently be redistributed and/or involve the use of third party software.

CLASSPATH EXCEPTION
Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

1.178 spring-retry 1.3.1
1.178.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2006-2019 the original author or authors.
 * *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * *
 * https://www.apache.org/licenses/LICENSE-2.0
 * *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
/*
* Copyright 2002-2008 the original author or authors.
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
*      https://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources-jar/org/springframework/retry/policy/BinaryExceptionClassifierRetryPolicy.java
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources-jar/org/springframework/classify/BinaryExceptionClassifier.java
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources-jar/org/springframework/retry/policy/MaxAttemptsRetryPolicy.java
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources-jar/org/springframework/retry/listener/MethodInvocationRetryListenerSupport.java
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources-jar/org/springframework/retry/interceptor/MethodInvocationRetryCallback.java
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources-jar/org/springframework/retry/policy/SimpleRetryPolicy.java
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources-jar/org/springframework/retry/interceptor/StatefulRetryOperationsInterceptor.java

No license file was found, but licenses were detected in source scan.

/ *
* Copyright 2014-2018 the original author or authors.
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
*/

Found in path(s):
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources-jar/org/springframework/classify/util/MethodInvoker.java
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources-jar/org/springframework/classify/util/AnnotationMethodResolver.java
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources-jar/org/springframework/classify/util/MethodResolver.java

No license file was found, but licenses were detected in source scan.
* You may obtain a copy of the License at
* https://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources-jar/org/springframework/retry/annotation/AnnotationAwareRetryOperationsInterceptor.java

No license file was found, but licenses were detected in source scan.

*/

* Copyright 2006-2020 the original author or authors.
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources-jar/org/springframework/retry/support/RetryTemplate.java
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources-jar/org/springframework/retry/backoff/ExponentialBackOffPolicy.java

No license file was found, but licenses were detected in source scan.

*/

* Copyright 2014 the original author or authors.
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources-jar/org/springframework/retry/backoff/ThreadWaitSleeper.java
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources-jar/org/springframework/retry/interceptor/RetryInterceptorBuilder.java
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources-jar/org/springframework/classify/util/SimpleMethodInvoker.java
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources-jar/org/springframework/retry/annotation/RetryConfiguration.java
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources-jar/org/springframework/retry/interceptor/Retryable.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2006-2014 the original author or authors.
* *
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* *
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources-jar/org/springframework/retry/backoff/FixedBackOffPolicy.java
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources-jar/org/springframework/retry/context/RetryContextSupport.java
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources-jar/org/springframework/retry/backoff/UniformRandomBackOffPolicy.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2006-2007 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources-jar/org/springframework/retry/support/RetrySimulator.java
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources-jar/org/springframework/retry/backoff/BackOffInterruptedException.java
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources-jar/org/springframework/retry/listener/RetryListenerSupport.java
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources-jar/org/springframework/classify/PatternMatcher.java
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources-jar/org/springframework/retry/policy/CompositeRetryPolicy.java
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources-jar/org/springframework/classify/ClassifierAdapter.java
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources-jar/org/springframework/retry/RetryState.java
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources-jar/org/springframework/classify/Classifier.java
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources-jar/org/springframework/retry/interceptor/NewMethodArgumentsIdentifier.java
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources-jar/org/springframework/retry/policy/ExhaustedRetryException.java
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources-jar/org/springframework/retry/interceptor/MethodArgumentsKeyGenerator.java
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources-jar/org/springframework/retry/policy/RetryCacheCapacityExceededException.java
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources-jar/org/springframework/retry/policy/NeverRetryPolicy.java
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources-jar/org/springframework/retry/backoff/NoBackOffPolicy.java
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources-jar/org/springframework/retry/recovery/RecoveryCallback.java
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources.jar/org/springframework/retry/RetryListener.java
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources.jar/org/springframework/retry/TerminatedRetryException.java
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources.jar/org/springframework/retry/interceptor/MethodInvocationRecoverer.java
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources.jar/org/springframework/classify/PatternMatchingClassifier.java
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources.jar/org/springframework/retry/RetryPolicy.java
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources.jar/org/springframework/retry/RetryStatistics.java
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources.jar/org/springframework/retry/policy/SoftReferenceMapRetryContextCache.java
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources.jar/org/springframework/classify/ClassifierSupport.java
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources.jar/org/springframework/retry/support/DefaultRetryState.java
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources.jar/org/springframework/retry/backoff/BackOffContext.java
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources.jar/org/springframework/retry/RetryOperations.java
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources.jar/org/springframework/retry/RetryCallback.java
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources.jar/org/springframework/retry/backoff/SleepingBackOffPolicy.java
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources.jar/org/springframework/retry/policy/ExceptionClassifierRetryPolicy.java
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources.jar/org/springframework/classify/annotation/Classifier.java
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources.jar/org/springframework/classify/BackToBackPatternClassifier.java
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources.jar/org/springframework/retry/backoff/Sleeper.java
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources.jar/org/springframework/retry/backoff/StatelessBackOffPolicy.java
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources.jar/org/springframework/retry/policy/TimeoutRetryPolicy.java
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources.jar/org/springframework/retry/policy/RetryContextCache.java
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources.jar/org/springframework/retry/policy/RetryException.java
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources.jar/org/springframework/retry/policy/MapRetryContextCache.java
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources.jar/org/springframework/retry/support/RetrySynchronizationManager.java
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources.jar/org/springframework/retry/policy/AlwaysRetryPolicy.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014-2019 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources-jar/org/springframework/retry/annotation/Retryable.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015-2019 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources-jar/org/springframework/retry/support/RetryTemplateBuilder.java
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources-jar/org/springframework/retry/policy/ExpressionRetryPolicy.java
No license file was found, but licenses were detected in source scan.

/ *
* Copyright 2013-2019 the original author or authors.
* *
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* *
* https://www.apache.org/licenses/LICENSE-2.0
* *
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources-
jar/org/springframework/retry/annotation/RecoverAnnotationRecoveryHandler.java

No license file was found, but licenses were detected in source scan.

/ *
* Copyright 2012-2013 the original author or authors.
* *
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* *
* https://www.apache.org/licenses/LICENSE-2.0
* *
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources-
jar/org/springframework/retry/annotation/EnableRetry.java
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources-
jar/org/springframework/retry/annotation/Backoff.java
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources-
jar/org/springframework/retry/annotation/Recover.java

No license file was found, but licenses were detected in source scan.
/*
 * Copyright 2006-2013 the original author or authors.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 *      https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources-jar/org/springframework/classify/SubclassClassifier.java
   No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2014 the original author or authors.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 *      https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources-jar/org/springframework/classify/util/MethodInvokerUtils.java
   No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2019 the original author or authors.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 */
* https://www.apache.org/licenses/LICENSE-2.0

* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources-jar/org/springframework/classify/BinaryExceptionClassifierBuilder.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2006-2012 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *   https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources-jar/org/springframework/retry/backoff/ExponentialRandomBackOffPolicy.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *   https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources-
jar/org/springframework/retry/policy/CircuitBreakerRetryPolicy.java
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources-
jar/org/springframework/retry/interceptor/FixedKeyGenerator.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2006-2007 the original author or authors.
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with
* the License. You may obtain a copy of the License at
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on
* an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources-
jar/org/springframework/retry/interceptor/RetryOperationsInterceptor.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2012-2015 the original author or authors.
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with
* the License. You may obtain a copy of the License at
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on
* an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and limitations under the License.
*/

Found in path(s):
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2016-2018 the original author or authors.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * https://www.apache.org/licenses/LICENSE-2.0
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1257203688_1643390885.66/0/spring-retry-1-3-1-sources-jar/org/springframework/retry/annotation/CircuitBreaker.java

1.179 antlr 2.7.7
1.179.1 Available under license:
SOFTWARE RIGHTS
ANTLR 1989-2006 Developed by Terence Parr
Partially supported by University of San Francisco & jGuru.com

We reserve no legal rights to the ANTLR--it is fully in the
public domain. An individual or company may do whatever they wish with source code distributed with ANTLR or the code generated by ANTLR, including the incorporation of ANTLR, or its output, into commercial software.

We encourage users to develop software with ANTLR. However, we do ask that credit is given to us for developing ANTLR. By "credit", we mean that if you use ANTLR or incorporate any source code into one of your programs (commercial product, research project, or otherwise) that you acknowledge this fact somewhere in the documentation, research report, etc... If you like ANTLR and have developed a nice tool with the output, please mention that you developed it using ANTLR. In addition, we ask that the headers remain intact in our source code. As long as these guidelines are kept, we expect to continue enhancing this system and expect to make other tools available as they are completed.

The primary ANTLR guy:

Terence Parr
parrt@cs.usfca.edu
parrt@antlr.org

1.180 httpcomponents-core 4.4.14

1.180.1 Available under license:

Apache HttpComponents Core
Copyright 2005-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.
"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."
"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

Apache HttpComponents Core
Copyright 2005-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.
"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.181 args4j 2.0.29

1.182 netty-reactive-streams 2.0.4
1.182.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner.
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
   Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
    stating that You changed the files; and
(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0
Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.183 jsr305 3.0.2

1.183.1 Available under license:
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2005 Brian Goetz
 * Released under the Creative Commons Attribution License
 * (http://creativecommons.org/licenses/by/2.5)
 * Official home: http://www.jcip.net
 */

Found in path(s):
* /opt/cola/permits/1135861076_1613617584.8/0/jsr305-3-0-2-sources-4-
jar/javax/annotation/concurrent/GuardedBy.java
* /opt/cola/permits/1135861076_1613617584.8/0/jsr305-3-0-2-sources-4-
jar/javax/annotation/concurrent/ThreadSafe.java
* /opt/cola/permits/1135861076_1613617584.8/0/jsr305-3-0-2-sources-4-
jar/javax/annotation/concurrent/Immutable.java
* /opt/cola/permits/1135861076_1613617584.8/0/jsr305-3-0-2-sources-4-
jar/javax/annotation/concurrent/NotThreadSafe.java

1.184 kotlin 1.4.32

1.184.1 Available under license:
No license file was found, but licenses were detected in source scan.

{"version":3,"file":"kotlin.js","sources":[]}
Kotlin.toChar(Kotlin.numberToChar(a));

Kotlin.doubleToInt = function(a) {
    if (a > 2147483647) return 2147483647;
    if (a < -2147483648) return -2147483648;
    return a | 0;
};

Kotlin.unboxChar = function(a) {
    if (a == null) return a;
    if (a instanceof Kotlin.BoxedChar) return a;
    return new Kotlin.BoxedChar(a);
};

Kotlin.equals = function (obj1, obj2) {
    if (obj1 == null) {
        return obj2 == null;
    }

    if (obj2 == null) {
        return false;
    }

    if (obj1 !== obj1) {
        return obj2 !== obj2;
    }

    if (typeof obj1 === "object" && typeof obj1.equals === "function") {
        return obj1.equals(obj2);
    }

    if (typeof obj1 === "number" && typeof obj2 === "number") {
        return obj1 === obj2 && (obj1 !== 0 || 1 / obj1 === 1 / obj2);
    }

    return obj1 === obj2;
};

Kotlin.hashCode = function (obj) {
    if (obj == null) {
        return 0;
    }

    var objType = typeof obj;
    if ("object" === objType) {
        return "function" === typeof obj.hashCode ? obj.hashCode() : getHashCode(obj);
    }

    if ("function" === objType) {
        return getHashCode(obj);
    }

    if ("number" === objType) {
        return Kotlin.numberHashCode(obj);
    }

    var str = String(obj);
    return getStringHashCode(str);
}

Kotlin.toString = function (o) {
    if (o == null) {
        return "null";
    }

    if (Kotlin.isArrayish(o)) {
        return "[...]";
    }

    return o.toString();
};

/** @const */
var POW_2_32 = 4294967296;
/** @const */
var OBJECT_HASH_CODE_PROPERTY_NAME = "kotlinHashCodeValue$";

function getHashCode(obj) {
    if (!(OBJECT_HASH_CODE_PROPERTY_NAME in obj)) {
        var hash = (Math.random() * POW_2_32) | 0; // Make 32-bit unique:
        Object.defineProperty(obj, OBJECT_HASH_CODE_PROPERTY_NAME, {
            value: hash, enumerable: false });
    }

    return obj[OBJECT_HASH_CODE_PROPERTY_NAME];
}

Kotlin.identityHashCode = getHashCode;

of the Long representations of small integer values.

`Kotlin.Long.IntCache_ = {};

Kotlin.Long.fromInt = function(value) {
  if (-128 <= value && value < 128) {
    var cachedObj = Kotlin.Long.IntCache_[value];
    if (cachedObj) {
      return cachedObj;
    }
  }

  var obj = new Kotlin.Long(value | 0, value < 0 ? -1 : 0);
  if (-128 <= value && value < 128) {
  }
  return obj;
};

Kotlin.Long.fromNumber = function(value) {
  if (isNaN(value)) {
    return Kotlin.Long.ZERO;
  } else if (value <= -Kotlin.Long.TWO_PWR_63_DBL_) {
    return Kotlin.Long.MIN_VALUE;
  } else if (value + 1 >= Kotlin.Long.TWO_PWR_63_DBL_) {
    return Kotlin.Long.MAX_VALUE;
  } else if (value < 0) {
    return Kotlin.Long.fromNumber(-value).negate();
  } else {
    return new Kotlin.Long((value % Kotlin.Long.TWO_PWR_32_DBL_) | 0,
        (value / Kotlin.Long.TWO_PWR_32_DBL_) | 0);
  }
};

Kotlin.Long.TWO_PWR_16_DBL_ = 1 << 16;

Kotlin.Long.TWO_PWR_24_DBL_ = 1 << 24;


Kotlin.Long.TWO_PWR_31_DBL_ = Kotlin.Long.TWO_PWR_32_DBL_ / 2;

Kotlin.Long.TWO_PWR_64_DBL_ = Kotlin.Long.TWO_PWR_32_DBL_ * 2;

Kotlin.Long.ONE = Kotlin.Long.fromInt(1);

Kotlin.Long.ZERO = Kotlin.Long.fromInt(0);

Kotlin.Long.MAX_VALUE = Kotlin.Long.fromInt(2147483647);

Kotlin.Long.MIN_VALUE = Kotlin.Long.fromInt(-2147483648);
Kotlin.Long.fromBits(0xFFFFFFFF | 0, 0x7FFFFFFF | 0);\n\n/**
 * @type {!Kotlin.Long}
 */
Kotlin.Long.MIN_VALUE = Kotlin.Long.fromBits(0, 0x80000000 | 0);\n\n/**
 * @type {!Kotlin.Long}
 */
Kotlin.Long.TWO_PWR_24_ = Kotlin.Long.fromInt(1 << 24);\n\n/**
 * @return {number}
 */
Kotlin.Long.MIN_VALUE = Kotlin.Long.fromBits(0, 0x80000000 | 0);\n\n/**
 * @type {!Kotlin.Long}
 *
 * @private
 */
Kotlin.Long.TWO_PWR_24_ = Kotlin.Long.fromInt(1 << 24);\n\n/**
 * @return {number}
 */
Kotlin.Long.prototype.toInt = function() {\n  return this.low_ &\n\n/**
 * @return {number}
 */
Kotlin.Long.prototype.toNumber = function() {\n  return this.high_ *\n  \n.getLowBitsUnsigned() &\n\n/**
 * @return {number}
 */
Kotlin.Long.prototype.toNumber = function() {\n  return this.high_ *\n  \n.TWO_PWR_32_DBL_ +\n\n*/\nKotlin.Long.prototype.toString = function(opt_radix) {\n  var radix = opt_radix || 10;\n  if (radix < 2 || 36 < radix) {\n    throw Error('radix out of range: ' + radix);\n  }\n\n  if (this.isZero()) {\n    return '0';\n  }\n\n  if (this.isNegative()) {\n    if (this.equalsLong(Kotlin.Long.MIN_VALUE)) {\n      // We need to change the Long value before it can be negated, so we remove\n      // the bottom-most digit in this base and then use\n      // recurse to do the rest.\n      var radixLong = Kotlin.Long.fromNumber(radix);\n      var div = this.div(radixLong);\n      var rem = div.multiply(radixLong).subtract(this);\n      return div.toString(radix) + rem.toInt().toString(radix);\n    }\n    else {\n      return '-' + this.negate().toString(radix);\n    }\n  }\n\n  // Do several (6) digits each time through the loop,\n  // so as to\n  // minimize the calls to the very expensive emulated div.\n  var radixToPower =\n  \n.getHighBits = function() {\n    return this.high_;\n  }\n\n/**
 * @return {number}
 */
Kotlin.Long.prototype.getLowBits = function() {\n  return this.low_;\n}
equal to the other.

```javascript
Kotlin.Long.prototype.greaterThanOrEqual = function(other) {
  return this.compare(other) >= 0;
}

Kotlin.Long.prototype.greaterThanOrEqual = function(other) {
  return this.compare(other) >= 0;
}
```
```java
// Repeat the following until the remainder is less than other: find an
// floating-point that approximates remainder / other *from below*, add this
// into the result, and subtract it from the remainder. It is critical that
// the approximate value is less than or equal to the real value so that the
// remainder never becomes negative.
var res = Kotlin.Long.ZERO;
while (rem.greaterThanOrEqual(other)) {
    // Approximate the result of division. This may be a little greater or
    // smaller than the actual value.
    var approx = Math.max(1, Math.floor(rem.toNumber() / other.toNumber()));

    // We know the answer can’t be zero... and actually, zero would cause
    // infinite recursion since we would make no progress.
    if (approxRes.isZero()) {
        approxRes = Kotlin.Long.ONE;
    }

    res = res.add(approxRes);
    rem = rem.subtract(approxRem);
}

return res;
```
of bits by which to shift.

```javascript
* @return {!Kotlin.Long}
This shifted to the right by the given amount, with
zeros placed into the new leading bits.

```
(bufInt32[lowIndex] !== 0) {\n
lowIndex = 1;\n
highIndex = 0;\n }\n\n Kotlin.doubleToBits =
function(value) {\n
return Kotlin.doubleToRawBits(isNaN(value) ? NaN : value);\n };\n\n
Kotlin.doubleToRawBits = function(value) {\n
bufFloat64[0] = value;\n
return
Kotlin.Long.fromBits(bufInt32[lowIndex], bufInt32[highIndex]);\n };\n\n Kotlin.doubleFromBits =
function(value) {\n
bufInt32[lowIndex] = value.low_;\n
bufInt32[highIndex] = value.high_;\n
return
bufFloat64[0];\n };\n\n Kotlin.floatToBits = function(value) {\n
return Kotlin.floatToRawBits(isNaN(value)
? NaN : value);\n };\n\n Kotlin.floatToRawBits = function(value) {\n
bufFloat32[0] = value;\n
return
bufInt32[0];\n };\n\n Kotlin.floatFromBits = function(value) {\n
bufInt32[0] = value;\n
return
bufFloat32[0];\n };\n\n // returns zero value for number with positive sign bit and non-zero value for number
with negative sign bit.\n Kotlin.doubleSignBit = function(value) {\n
bufFloat64[0] = value;\n
return
bufInt32[highIndex] & 0x80000000;\n };\n\n Kotlin.numberHashCode = function(obj) {\n
if ((obj | 0) ===
obj) {\n
return obj | 0;\n
}\n
else {\n
bufFloat64[0] = obj;\n
return (bufInt32[highIndex]
* 31 | 0) + bufInt32[lowIndex] | 0;\n
}\n }\n})();\n\nKotlin.ensureNotNull = function(x) {\n return x != null
? x : Kotlin.throwNPE();\n};\n","/*\n * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language
contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the
license/LICENSE.txt file.\n */\n\nif (typeof String.prototype.startsWith === \"undefined\") {\n
Object.defineProperty(String.prototype, \"startsWith\", {\n
value: function (searchString, position) {\n
position = position || 0;\n
return this.lastIndexOf(searchString, position) === position;\n
}\n });\n}\nif
(typeof String.prototype.endsWith === \"undefined\") {\n Object.defineProperty(String.prototype, \"endsWith\",
{\n
value: function (searchString, position) {\n
var subjectString = this.toString();\n
if (position
=== undefined || position > subjectString.length) {\n
position = subjectString.length;\n
}\n
position -= searchString.length;\n
var lastIndex = subjectString.indexOf(searchString, position);\n
return lastIndex !== -1 && lastIndex === position;\n
}\n });\n}\n// ES6 Math polyfills\nif (typeof Math.sign
=== \"undefined\") {\n Math.sign = function(x) {\n
x = +x; // convert to a number\n
if (x === 0 ||
isNaN(x)) {\n
return Number(x);\n
}\n
return x > 0 ? 1 : -1;\n };\n}\nif (typeof Math.trunc ===
\"undefined\") {\n Math.trunc = function(x) {\n
if (isNaN(x)) {\n
return NaN;\n
}\n
if (x > 0)
{\n
return Math.floor(x);\n
}\n
return Math.ceil(x);\n };\n}\n\n(function() {\n var epsilon =
2.220446049250313E-16;\n var taylor_2_bound = Math.sqrt(epsilon);\n var taylor_n_bound =
Math.sqrt(taylor_2_bound);\n var upper_taylor_2_bound = 1/taylor_2_bound;\n var upper_taylor_n_bound =
1/taylor_n_bound;\n\n if (typeof Math.sinh === \"undefined\") {\n
Math.sinh = function(x) {\n
if
(Math.abs(x) < taylor_n_bound) {\n
var result = x;\n
if (Math.abs(x) > taylor_2_bound) {\n
result += (x * x * x) / 6;\n
}\n
return result;\n
} else {\n
var y =
Math.exp(x);\n
var y1 = 1 / y;\n
if (!isFinite(y)) return Math.exp(x - Math.LN2);\n
if
(!isFinite(y1)) return -Math.exp(-x - Math.LN2);\n
return (y - y1) / 2;\n
}\n
};\n }\n if
(typeof Math.cosh === \"undefined\") {\n
Math.cosh = function(x) {\n
var y = Math.exp(x);\n
var
y1 = 1 / y;\n
if (!isFinite(y) || !isFinite(y1)) return Math.exp(Math.abs(x) - Math.LN2);\n
return (y +
y1) / 2;\n
};\n }\n\n if (typeof Math.tanh === \"undefined\") {\n
Math.tanh = function(x){\n
if
(Math.abs(x) < taylor_n_bound) {\n
var result = x;\n
if (Math.abs(x) > taylor_2_bound) {\n
result -= (x * x * x) / 3;\n
}\n
return result;\n
}\n
else {\n
var a =
Math.exp(+x), b = Math.exp(-x);\n
return a === Infinity ? 1 : b === Infinity ? -1 : (a - b) / (a + b);\n
}\n
};\n }\n\n // Inverse hyperbolic function implementations derived from boost special math functions,\n
// Copyright Eric Ford & Hubert Holin 2001.\n\n if (typeof Math.asinh === \"undefined\") {\n
var asinh =
function(x) {\n
if (x >= +taylor_n_bound)\n
{\n
if (x > upper_taylor_n_bound)\n
{\n
if (x > upper_taylor_2_bound)\n
{\n
// approximation by laurent series in
1/x at 0+ order from -1 to 0\n
return Math.log(x) + Math.LN2;\n
}\n
else\n
{\n
// approximation by laurent series in 1/x at 0+ order from -1 to 1\n
return
Math.log(x * 2 + (1 / (x * 2)));\n
}\n
}\n
else\n
{\n
return
Math.log(x + Math.sqrt(x * x + 1));\n
}\n
}\n
else if (x <= -taylor_n_bound)\n
{\n

Open Source Used In security_automation identity-guest-cs 1169


if (typeof TypedArray.prototype.fill === "undefined") {
    Object.defineProperty(TypedArray.prototype, 'fill', {
        value: Array.prototype.fill
    });
}
if (typeof TypedArray.prototype.slice === "undefined") {
    Object.defineProperty(TypedArray.prototype, 'slice', {
        value: typedArraySlice
    });
}
if (typeof TypedArray.prototype.map === "undefined") {
    Object.defineProperty(TypedArray.prototype, 'map', {
        value: function(callback, self) {
            return [].slice.call(this).map(callback, self);
        }
    });
}
if (typeof TypedArray.prototype.sort === "undefined") {
    Object.defineProperty(TypedArray.prototype, 'sort', {
        value: function(compareFunction) {
            return Array.prototype.sort.call(this, compareFunction);
        }
    });
}

TypedArray = arrays[i];
if (typeof TypedArray.prototype.fill === "undefined") {
    Object.defineProperty(TypedArray.prototype, 'fill', {
        value: Array.prototype.fill
    });
}
if (typeof TypedArray.prototype.slice === "undefined") {
    Object.defineProperty(TypedArray.prototype, 'slice', {
        value: typedArraySlice
    });
}
if (typeof TypedArray.prototype.map === "undefined") {
    Object.defineProperty(TypedArray.prototype, 'map', {
        value: function(callback, self) {
            return [].slice.call(this).map(callback, self);
        }
    });
}
if (typeof TypedArray.prototype.sort === "undefined") {
    Object.defineProperty(TypedArray.prototype, 'sort', {
        value: function(compareFunction) {
            return Array.prototype.sort.call(this, compareFunction);
        }
    });
}

TypedArray = arrays[i];
DOM `typeof` returns `"object"`, nevertheless they can be used in RHS of `instanceof` if `(klassMetadata == null)` `{ return object instanceof klass; }` if `(klassMetadata.kind === Kotlin.KindINTERFACE && object.constructor != null)` `{ return isInheritanceFromInterface(object.constructor, klass); }` return false; } `\n\nKotlin.isNumber = function (a) { return typeof a === 'number'; }` `\n\nKotlin.isChar = function (value) { return value instanceof Kotlin.BoxedChar; }` `\n\nKotlin.isComparable = function (value) { return typeof value === 'string' || typeof value === 'boolean' || Kotlin.isNumber(value) || Kotlin.isType(value, Kotlin.kotlin.Comparable); }` `\n\nKotlin.isCharSequence = function (value) { return typeof value === 'string' || Kotlin.isType(value, Kotlin.kotlin.CharSequence); }` "*/ Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors. * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file. */ A package is omitted to get declarations directly under the module `@PublishedApi` `fun <T> fillArrayFun(array: Array<T>, init: (Int) -> T) { for (i in 0..array.size - 1) { array[i] = init(i); } return array; }` `\n\nKotlin.isNumber = function (a) { return typeof a == 'number' || a instanceof Kotlin.Long; }` `\n\nKotlin.isChar = function (value) { return value instanceof Kotlin.BoxedChar; }` `\n\nKotlin.isComparable = function (value) { var type = typeof value; return type === 'string' || typeof value === 'boolean' || Kotlin.isNumber(value) || Kotlin.isType(value, Kotlin.kotlin.Comparable); }` `\n\nKotlin.isCharSequence = function (value) { return typeof value === 'string' || Kotlin.isType(value, Kotlin.kotlin.CharSequence); }` "*/ Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors. * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file. */ A package is omitted to get declarations directly under the module `@PublishedApi` `fun <T> fillArrayFun(array: Array<T>, init: (Int) -> T) { for (i in 0..array.size - 1) { array[i] = init(i); } return array; }` `\n\nKotlin.isNumber = function (a) { return typeof a == 'number' || a instanceof Kotlin.Long; }` `\n\nKotlin.isChar = function (value) { return value instanceof Kotlin.BoxedChar; }` `\n\nKotlin.isComparable = function (value) { var type = typeof value; return type === 'string' || typeof value === 'boolean' || Kotlin.isNumber(value) || Kotlin.isType(value, Kotlin.kotlin.Comparable); }` `\n\nKotlin.isCharSequence = function (value) { return typeof value === 'string' || Kotlin.isType(value, Kotlin.kotlin.CharSequence); }` "*/ Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors. * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file. */ A package is omitted to get declarations directly under the module `@PublishedApi` `fun <T> fillArrayFun(array: Array<T>, init: (Int) -> T) { for (i in 0..array.size - 1) { array[i] = init(i); } return array; }` `\n\nKotlin.isNumber = function (a) { return typeof a == 'number' || a instanceof Kotlin.Long; }` `\n\nKotlin.isChar = function (value) { return value instanceof Kotlin.BoxedChar; }` `\n\nKotlin.isComparable = function (value) { var type = typeof value; return type === 'string' || typeof value === 'boolean' || Kotlin.isNumber(value) || Kotlin.isType(value, Kotlin.kotlin.Comparable); }` `\n\nKotlin.isCharSequence = function (value) { return typeof value === 'string' || Kotlin.isType(value, Kotlin.kotlin.CharSequence); }` "*/ Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors. * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file. */ A package is omitted to get declarations directly under the module `@PublishedApi` `fun <T> fillArrayFun(array: Array<T>, init: (Int) -> T) { for (i in 0..array.size - 1) { array[i] = init(i); } return array; }` `\n\nKotlin.isNumber = function (a) { return typeof a == 'number' || a instanceof Kotlin.Long; }` `\n\nKotlin.isChar = function (value) { return value instanceof Kotlin.BoxedChar; }` `\n\nKotlin.isComparable = function (value) { var type = typeof value; return type === 'string' || typeof value === 'boolean' || Kotlin.isNumber(value) || Kotlin.isType(value, Kotlin.kotlin.Comparable); }` `\n\nKotlin.isCharSequence = function (value) { return typeof value === 'string' || Kotlin.isType(value, Kotlin.kotlin.CharSequence); }"
kotlin.ranges.contains\n
import kotlin.ranges.reversed\n
/**
 * Returns 1st *element* from the array.
 * If the size of this array is less than 1, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@kotlin.internal.InlineOnly\npublic inline operator fun <T> Array<T>.component1(): T {
    return get(0)
}

/**
 * Returns 1st *element* from the array.
 * If the size of this array is less than 1, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@kotlin.internal.InlineOnly\npublic inline operator fun ByteArray.component1(): Byte {
    return get(0)
}

/**
 * Returns 1st *element* from the array.
 * If the size of this array is less than 1, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@kotlin.internal.InlineOnly\npublic inline operator fun ShortArray.component1(): Short {
    return get(0)
}

/**
 * Returns 1st *element* from the array.
 * If the size of this array is less than 1, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@kotlin.internal.InlineOnly\npublic inline operator fun IntArray.component1(): Int {
    return get(0)
}

/**
 * Returns 1st *element* from the array.
 * If the size of this array is less than 1, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@kotlin.internal.InlineOnly\npublic inline operator fun LongArray.component1(): Long {
    return get(0)
}

/**
 * Returns 1st *element* from the array.
 * If the size of this array is less than 1, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@kotlin.internal.InlineOnly\npublic inline operator fun FloatArray.component1(): Float {
    return get(0)
}

/**
 * Returns 1st *element* from the array.
 * If the size of this array is less than 1, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@kotlin.internal.InlineOnly\npublic inline operator fun DoubleArray.component1(): Double {
    return get(0)
}

/**
 * Returns 1st *element* from the array.
 * If the size of this array is less than 1, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@kotlin.internal.InlineOnly\npublic inline operator fun BooleanArray.component1(): Boolean {
    return get(0)
}

/**
 * Returns 1st *element* from the array.
 * If the size of this array is less than 1, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@kotlin.internal.InlineOnly\npublic inline operator fun CharArray.component1(): Char {
    return get(0)
}

/**
 * Returns 2nd *element* from the array.
 * If the size of this array is less than 2, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@kotlin.internal.InlineOnly\npublic inline operator fun <T> Array<T>.component2(): T {
    return get(1)
}

/**
 * Returns 2nd *element* from the array.
 * If the size of this array is less than 2, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@kotlin.internal.InlineOnly\npublic inline operator fun ByteArray.component2(): Byte {
    return get(1)
}

/**
 * Returns 2nd *element* from the array.
 * If the size of this array is less than 2, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@kotlin.internal.InlineOnly\npublic inline operator fun ShortArray.component2(): Short {
    return get(1)
}

/**
 * Returns 2nd *element* from the array.
 * If the size of this array is less than 2, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@kotlin.internal.InlineOnly\npublic inline operator fun IntArray.component2(): Int {
    return get(1)
}

/**
 * Returns 2nd *element* from the array.
 * If the size of this array is less than 2, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@kotlin.internal.InlineOnly\npublic inline operator fun LongArray.component2(): Long {
    return get(1)
}

/**
 * Returns 2nd *element* from the array.
 * If the size of this array is less than 2, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@kotlin.internal.InlineOnly\npublic inline operator fun FloatArray.component2(): Float {
    return get(1)
}

/**
 * Returns 2nd *element* from the array.
 * If the size of this array is less than 2, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@kotlin.internal.InlineOnly\npublic inline operator fun DoubleArray.component2(): Double {
    return get(1)
}
get(1)

* Returns 2nd *element* from the array.

* If the size of this array is less than 2, throws an
[IndexOutOfBoundsException] except in Kotlin/JS
* where the behavior is unspecified.

* @kotlin.internal.InlineOnly
*public inline operator fun BooleanArray.component2(): Boolean {
* return get(1)

* Returns 2nd *element* from the array.
* If the size of this array is less than 2, throws an
[IndexOutOfBoundsException] except in Kotlin/JS
* where the behavior is unspecified.

* @kotlin.internal.InlineOnly
*public inline operator fun CharArray.component2(): Char {
* return get(1)

* Returns 2nd *element* from the array.
* If the size of this array is less than 2, throws an
[IndexOutOfBoundsException] except in Kotlin/JS
* where the behavior is unspecified.

* @kotlin.internal.InlineOnly
*public inline operator fun ShortArray.component2(): Short {
* return get(2)

* Returns 3rd *element* from the array.
* If the size of this array is less than 3, throws an
[IndexOutOfBoundsException] except in Kotlin/JS
* where the behavior is unspecified.

* @kotlin.internal.InlineOnly
*public inline operator fun IntArray.component2(): Int {
* return get(2)

* Returns 3rd *element* from the array.
* If the size of this array is less than 3, throws an
[IndexOutOfBoundsException] except in Kotlin/JS
* where the behavior is unspecified.

* @kotlin.internal.InlineOnly
*public inline operator fun LongArray.component2(): Long {
* return get(2)

* Returns 3rd *element* from the array.
* If the size of this array is less than 3, throws an
[IndexOutOfBoundsException] except in Kotlin/JS
* where the behavior is unspecified.

* @kotlin.internal.InlineOnly
*public inline operator fun DoubleArray.component2(): Double {
* return get(2)

* Returns 3rd *element* from the array.
* If the size of this array is less than 3, throws an
[IndexOutOfBoundsException] except in Kotlin/JS
* where the behavior is unspecified.

* @kotlin.internal.InlineOnly
*public inline operator fun BooleanArray.component3(): Boolean {
* return get(2)

* Returns 3rd *element* from the array.
* If the size of this array is less than 3, throws an
[IndexOutOfBoundsException] except in Kotlin/JS
* where the behavior is unspecified.

* @kotlin.internal.InlineOnly
*public inline operator fun CharArray.component3(): Char {
* return get(2)

* Returns 3rd *element* from the array.
* If the size of this array is less than 3, throws an
[IndexOutOfBoundsException] except in Kotlin/JS
* where the behavior is unspecified.

* @kotlin.internal.InlineOnly
*public inline operator fun ShortArray.component3(): Short {
* return get(2)

* Returns 3rd *element* from the array.
* If the size of this array is less than 3, throws an
[IndexOutOfBoundsException] except in Kotlin/JS
* where the behavior is unspecified.

* @kotlin.internal.InlineOnly
*public inline operator fun IntArray.component3(): Int {
* return get(2)

* Returns 3rd *element* from the array.
* If the size of this array is less than 3, throws an
[IndexOutOfBoundsException] except in Kotlin/JS
* where the behavior is unspecified.

* @kotlin.internal.InlineOnly
*public inline operator fun LongArray.component3(): Long {
* return get(2)

* Returns 3rd *element* from the array.
* If the size of this array is less than 3, throws an
[IndexOutOfBoundsException] except in Kotlin/JS
* where the behavior is unspecified.

* @kotlin.internal.InlineOnly
*public inline operator fun DoubleArray.component3(): Double {
* return get(2)

* Returns 3rd *element* from the array.
* If the size of this array is less than 3, throws an
[IndexOutOfBoundsException] except in Kotlin/JS
* where the behavior is unspecified.

* @kotlin.internal.InlineOnly
*public inline operator fun BooleanArray.component4(): Boolean {
* return get(3)

* Returns 4th *element* from the array.
* If the size of this array is less than 4, throws an
[IndexOutOfBoundsException] except in Kotlin/JS
* where the behavior is unspecified.

* @kotlin.internal.InlineOnly
*public inline operator fun CharArray.component4(): Char {
* return get(3)

* Returns 4th *element* from the array.
* If the size of this array is less than 4, throws an
[IndexOutOfBoundsException] except in Kotlin/JS
* where the behavior is unspecified.

* @kotlin.internal.InlineOnly
*public inline operator fun ShortArray.component4(): Short {
* return get(3)

* Returns 4th *element* from the array.
* If the size of this array is less than 4, throws an
[IndexOutOfBoundsException] except in Kotlin/JS
* where the behavior is unspecified.

* @kotlin.internal.InlineOnly
*public inline operator fun IntArray.component4(): Int {
* return get(3)

* Returns 4th *element* from the array.
* If the size of this array is less than 4, throws an
[IndexOutOfBoundsException] except in Kotlin/JS
* where the behavior is unspecified.

* @kotlin.internal.InlineOnly
*public inline operator fun LongArray.component4(): Long {
* return get(3)

* Returns 4th *element* from the array.
* If the size of this array is less than 4, throws an
[IndexOutOfBoundsException] except in Kotlin/JS
* where the behavior is unspecified.

* @kotlin.internal.InlineOnly
*public inline operator fun DoubleArray.component4(): Double {
* return get(3)

* Returns 4th *element* from the array.
* If the size of this array is less than 4, throws an
[IndexOutOfBoundsException] except in Kotlin/JS
* where the behavior is unspecified.
get(3) * Returns 4th *element* from the array.

If the size of this array is less than 4, throws an
[IndexOutOfBoundsException] except in Kotlin/JS

where the behavior is unspecified.

@kotlin.internal.InlineOnly
public inline operator fun FloatArray.component4(): Float {
    return get(3)
}

/**
 * Returns 4th *element* from the array.
 * 
 * If the size of this array is less than 4, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.

@kotlin.internal.InlineOnly
public inline operator fun DoubleArray.component4(): Double {
    return get(3)
}

/**
 * Returns 4th *element* from the array.
 * 
 * If the size of this array is less than 4, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.

@kotlin.internal.InlineOnly
public inline operator fun BooleanArray.component4(): Boolean {
    return get(3)
}

/**
 * Returns 4th *element* from the array.
 * 
 * If the size of this array is less than 4, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.

@kotlin.internal.InlineOnly
public inline operator fun CharArray.component4(): Char {
    return get(3)
}

/**
 * Returns 5th *element* from the array.
 * 
 * If the size of this array is less than 5, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.

@kotlin.internal.InlineOnly
public inline operator fun <T> Array<out T>.component5(): T {
    return get(4)
}

/**
 * Returns 5th *element* from the array.
 * 
 * If the size of this array is less than 5, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.

@kotlin.internal.InlineOnly
public inline operator fun ByteArray.component5(): Byte {
    return get(4)
}

/**
 * Returns 5th *element* from the array.
 * 
 * If the size of this array is less than 5, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.

@kotlin.internal.InlineOnly
public inline operator fun ShortArray.component5(): Short {
    return get(4)
}

/**
 * Returns 5th *element* from the array.
 * 
 * If the size of this array is less than 5, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.

@kotlin.internal.InlineOnly
public inline operator fun IntArray.component5(): Int {
    return get(4)
}

/**
 * Returns 5th *element* from the array.
 * 
 * If the size of this array is less than 5, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.

@kotlin.internal.InlineOnly
public inline operator fun LongArray.component5(): Long {
    return get(4)
}

/**
 * Returns `true` if [element] is found in the array.
 */
public operator fun <T> Array<out T>.contains(element: T): Boolean {
    return indexOf(element) >= 0
}

/**
 * Returns `true` if [element] is found in the array.
 */
public operator fun ByteArray.contains(element: Byte): Boolean {
    return indexOf(element) >= 0
}

/**
 * Returns `true` if [element] is found in the array.
 */
public operator fun ShortArray.contains(element: Short): Boolean {
    return indexOf(element) >= 0
}

/**
 * Returns `true` if [element] is found in the array.
 */
public operator fun IntArray.contains(element: Int): Boolean {
    return indexOf(element) >= 0
}

/**
 * Returns `true` if [element] is found in the array.
 */
public operator fun LongArray.contains(element: Long): Boolean {
    return indexOf(element) >= 0
}

/**
 * Returns `true` if [element] is found in the array.
 */
public operator fun <@kotlin.internal.OnlyInputTypes T> Array<out T>.contains(element: T): Boolean {
    return indexOf(element) >= 0
}

/**
 * Returns `true` if [element] is found in the array.
 */
public operator fun ByteArray.contains(element: Byte): Boolean {
    return indexOf(element) >= 0
}

/**
 * Returns `true` if [element] is found in the array.
 */
public operator fun ShortArray.contains(element: Short): Boolean {
    return indexOf(element) >= 0
}

/**
 * Returns `true` if [element] is found in the array.
 */
public operator fun IntArray.contains(element: Int): Boolean {
    return indexOf(element) >= 0
}

/**
 * Returns `true` if [element] is found in the array.
 */
public operator fun LongArray.contains(element: Long): Boolean {
    return indexOf(element) >= 0
}

/**
 * Returns `true` if [element] is found in the array.
 */
public operator fun <@kotlin.internal.OnlyInputTypes T> Array<out T>.contains(element: T): Boolean {
    return indexOf(element) >= 0
}

/**
 * Returns `true` if [element] is found in the array.
 */
public operator fun ByteArray.contains(element: Byte): Boolean {
    return indexOf(element) >= 0
}

/**
 * Returns `true` if [element] is found in the array.
 */
public operator fun ShortArray.contains(element: Short): Boolean {
    return indexOf(element) >= 0
}

/**
 * Returns `true` if [element] is found in the array.
 */
public operator fun IntArray.contains(element: Int): Boolean {
    return indexOf(element) >= 0
}

/**
 * Returns `true` if [element] is found in the array.
 */
public operator fun LongArray.contains(element: Long): Boolean {
    return indexOf(element) >= 0
}

/**
 * Returns `true` if [element] is found in the array.
 */
@Deprecated("The

Open Source Used In security_automation identity-guest-cs 1175
The function has unclear behavior when searching for NaN or zero values and will be removed soon. Use `any { it == element }` instead to continue using this behavior, or `asList().contains(element: T)` to get the same search behavior as in a list.

```kotlin
@DeprecatedSinceKotlin(warningSince = "1.4")
@Suppress("DEPRECATION")
public operator fun FloatArray.contains(element: Float): Boolean {
    return indexOf(element) >= 0
}

/**
 * Returns `true` if [element] is found in the array.
 */
public operator fun DoubleArray.contains(element: Double): Boolean {
    return indexOf(element) >= 0
}

/**
 * Returns `true` if [element] is found in the array.
 */
public operator fun BooleanArray.contains(element: Boolean): Boolean {
    return indexOf(element) >= 0
}

/**
 * Returns `true` if [element] is found in the array.
 */
public operator fun CharArray.contains(element: Char): Boolean {
    return indexOf(element) >= 0
}

/**
 * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 */
public expect fun <T> Array<out T>.elementAt(index: Int): T

/**
 * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 */
public expect fun ByteArray.elementAt(index: Int): Byte

/**
 * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 */
public expect fun ShortArray.elementAt(index: Int): Short

/**
 * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 */
public expect fun IntArray.elementAt(index: Int): Int

/**
 * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 */
public expect fun LongArray.elementAt(index: Int): Long

/**
 * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 */
public expect fun FloatArray.elementAt(index: Int): Float

/**
 * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 */
public expect fun DoubleArray.elementAt(index: Int): Double

/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.
 */
@kotlin.internal.InlineOnly
public inline fun <T> Array<out T>.elementAtOrElse(index: Int, defaultValue: (Int) -> T): T {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}
```

Open Source Used In security_automation identity-guest-cs 1176
calling the [defaultValue] function if the [index] is out of bounds of this array.

```kotlin
*n
*n
*sample

samples.collections.Collections.Elements.elementAtOrElse(index: Int, defaultValue: (Int) -> Int): Int {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}
```

```kotlin
*n
*n
*sample

IntArray.elementAtOrElse(index: Int, defaultValue: (Int) -> Int): Int {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}
```

```kotlin
*n
*n
*sample

LongArray.elementAtOrElse(index: Int, defaultValue: (Int) -> Long): Long {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}
```

```kotlin
*n
*n
*sample

FloatArray.elementAtOrElse(index: Int, defaultValue: (Int) -> Float): Float {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}
```

```kotlin
*n
*n
*sample

CharArray.elementAtOrElse(index: Int, defaultValue: (Int) -> Char): Char {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}
```

```kotlin
*n
*n
*sample

ByteArray.elementAtOrNull(index: Int): Byte? {
    return this.getOrNull(index)
}
```

```kotlin
*n
*n
*sample

ShortArray.elementAtOrNull(index: Int): Short? {
    return this.getOrNull(index)
}
```

```kotlin
*n
*n
*sample

IntArray.elementAtOrNull(index: Int): Int? {
    return this.getOrNull(index)
}
```

```kotlin
*n
*n
*sample

LongArray.elementAtOrNull(index: Int): Long? {
    return this.getOrNull(index)
}
```

```kotlin
*n
*n
*sample

FloatArray.elementAtOrNull(index: Int): Float? {
    return this.getOrNull(index)
}
```

```kotlin
*n
*n
*sample

DoubleArray.elementAtOrNull(index: Int): Double? {
    return this.getOrNull(index)
}
```

```kotlin
*n
*n
*sample

BooleanArray.elementAtOrNull(index: Int): Boolean? {
    return this.getOrNull(index)
}
```

```kotlin
*n
*n
*sample

ByteArray.elementAtOrNull(index: Int): Byte? {
    return this.getOrNull(index)
}
```

```kotlin
*n
*n
*sample

ShortArray.elementAtOrNull(index: Int): Short? {
    return this.getOrNull(index)
}
```

```kotlin
*n
*n
*sample

IntArray.elementAtOrNull(index: Int): Int? {
    return this.getOrNull(index)
}
```

```kotlin
*n
*n
*sample

LongArray.elementAtOrNull(index: Int): Long? {
    return this.getOrNull(index)
}
```

```kotlin
*n
*n
*sample

FloatArray.elementAtOrNull(index: Int): Float? {
    return this.getOrNull(index)
}
```

```kotlin
*n
*n
*sample

DoubleArray.elementAtOrNull(index: Int): Double? {
    return this.getOrNull(index)
}
```

```kotlin
*n
*n
*sample

BooleanArray.elementAtOrNull(index: Int): Boolean? {
    return this.getOrNull(index)
}
```
element at the given [index] or 'null' if the [index] is out of bounds of this array.

```kotlin
CharArray.elementAtOrNull(index: Int): Char? {
    return this.getOrNull(index)
}
```

```kotlin
ByteArray.find(predicate: (Byte) -> Boolean): Byte? {
    return firstOrNull(predicate)
}
```

```kotlin
ShortArray.find(predicate: (Short) -> Boolean): Short? {
    return firstOrNull(predicate)
}
```

```kotlin
IntArray.find(predicate: (Int) -> Boolean): Int? {
    return firstOrNull(predicate)
}
```

```kotlin
LongArray.find(predicate: (Long) -> Boolean): Long? {
    return firstOrNull(predicate)
}
```

```kotlin
FloatArray.find(predicate: (Float) -> Boolean): Float? {
    return firstOrNull(predicate)
}
```

```kotlin
DoubleArray.find(predicate: (Double) -> Boolean): Double? {
    return firstOrNull(predicate)
}
```

```kotlin
BooleanArray.find(predicate: (Boolean) -> Boolean): Boolean? {
    return firstOrNull(predicate)
}
```

```kotlin
CharArray.find(predicate: (Char) -> Boolean): Char? {
    return firstOrNull(predicate)
}
```

```kotlin
ByteArray.findLast(predicate: (Byte) -> Boolean): Byte? {
    return lastOrNull(predicate)
}
```

```kotlin
ShortArray.findLast(predicate: (Short) -> Boolean): Short? {
    return lastOrNull(predicate)
}
```

```kotlin
IntArray.findLast(predicate: (Int) -> Boolean): Int? {
    return lastOrNull(predicate)
}
```

```kotlin
LongArray.findLast(predicate: (Long) -> Boolean): Long? {
    return lastOrNull(predicate)
}
```

```kotlin
FloatArray.findLast(predicate: (Float) -> Boolean): Float? {
    return lastOrNull(predicate)
}
```

```kotlin
DoubleArray.findLast(predicate: (Double) -> Boolean): Double? {
    return lastOrNull(predicate)
}
```

```kotlin
BooleanArray.findLast(predicate: (Boolean) -> Boolean): Boolean? {
    return lastOrNull(predicate)
}
```

```kotlin
CharArray.findLast(predicate: (Char) -> Boolean): Char? {
    return lastOrNull(predicate)
}
```

```kotlin
ByteArray.findLast(predicate: (Byte) -> Boolean): Byte? {
    return lastOrNull(predicate)
}
```

```kotlin
ShortArray.findLast(predicate: (Short) -> Boolean): Short? {
    return lastOrNull(predicate)
}
```

```kotlin
IntArray.findLast(predicate: (Int) -> Boolean): Int? {
    return lastOrNull(predicate)
}
```

```kotlin
LongArray.findLast(predicate: (Long) -> Boolean): Long? {
    return lastOrNull(predicate)
}
```

```kotlin
FloatArray.findLast(predicate: (Float) -> Boolean): Float? {
    return lastOrNull(predicate)
}
```

```kotlin
DoubleArray.findLast(predicate: (Double) -> Boolean): Double? {
    return lastOrNull(predicate)
}
```

```kotlin
BooleanArray.findLast(predicate: (Boolean) -> Boolean): Boolean? {
    return lastOrNull(predicate)
}
```

```kotlin
CharArray.findLast(predicate: (Char) -> Boolean): Char? {
    return lastOrNull(predicate)
}
```
the last element matching the given [predicate], or `null` if no such element was found.

```kotlin
public fun <T> Array<out T>.first(): T {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[0]
}
```

```
public fun ByteArray.first(): Byte {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[0]
}
```

```
public fun ShortArray.first(): Short {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[0]
}
```

```
public fun IntArray.first(): Int {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[0]
}
```

```
public fun LongArray.first(): Long {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[0]
}
```

```
public fun FloatArray.first(): Float {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[0]
}
```

```
public fun DoubleArray.first(): Double {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[0]
}
```

```
public fun BooleanArray.first(): Boolean {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[0]
}
```

```
public fun CharArray.first(): Char {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[0]
}
```

```
``
(predicate(element)) return element
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns the first element matching the given [predicate].
 * @throws [NoSuchElementException] if no such element is found.
 */
public inline fun DoubleArray.first(predicate: Double.() -> Boolean): Double {
    for (element in this) if (predicate(element)) return element
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns the first element matching the given [predicate].
 * @throws [NoSuchElementException] if no such element is found.
 */
public inline fun BooleanArray.first(predicate: Boolean.() -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return element
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns the first element matching the given [predicate].
 * @throws [NoSuchElementException] if no such element is found.
 */
public inline fun CharArray.first(predicate: Char.() -> Boolean): Char {
    for (element in this) if (predicate(element)) return element
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns the first element, or `null` if the array is empty.
 * @throws [NoSuchElementException] if no such element is found.
 */
public fun <T> Array<out T>.firstOrNull(): T? {
    return if (isEmpty()) null else this[0]
}

/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of
 */
public fun <T> Array<out T>.firstOrNull(index: Int, defaultValue: () -> T): T? {
    return if (index < 0 || index >= this.size) defaultValue() else this[index]
}

Open Source Used In security_automation identity-guest-cs 1180
bounds of this array.

@kotlin.internal.InlineOnly
public inline fun <T> Array<out T>.getOrElse(index: Int, defaultValue: (Int) -> T): T {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}

/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.
 */
@kotlin.internal.InlineOnly
public inline fun ByteArray.getOrElse(index: Int, defaultValue: (Int) -> Byte): Byte {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}

/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.
 */
@kotlin.internal.InlineOnly
public inline fun ShortArray.getOrElse(index: Int, defaultValue: (Int) -> Short): Short {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}

/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.
 */
@kotlin.internal.InlineOnly
public inline fun IntArray.getOrElse(index: Int, defaultValue: (Int) -> Int): Int {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}

/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.
 */
@kotlin.internal.InlineOnly
public inline fun LongArray.getOrElse(index: Int, defaultValue: (Int) -> Long): Long {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}

/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.
 */
@kotlin.internal.InlineOnly
public inline fun FloatArray.getOrElse(index: Int, defaultValue: (Int) -> Float): Float {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}

/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.
 */
@kotlin.internal.InlineOnly
public inline fun DoubleArray.getOrElse(index: Int, defaultValue: (Int) -> Double): Double {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}

/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
@kotlin.internal.InlineOnly
public inline fun CharArray.getOrElse(index: Int, defaultValue: (Int) -> Char): Char {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}

/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
@kotlin.internal.InlineOnly
public inline fun Array<out T>.getOrNull(index: Int): T? {
    return if (index >= 0 && index <= lastIndex) get(index) else null
}

/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
@kotlin.internal.InlineOnly
public inline fun ByteArray.getOrNull(index: Int): Byte? {
    return if (index >= 0 && index <= lastIndex) get(index) else null
}

/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
@kotlin.internal.InlineOnly
public inline fun ShortArray.getOrNull(index: Int): Short? {
    return if (index >= 0 && index <= lastIndex) get(index) else null
}

/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
@kotlin.internal.InlineOnly
public inline fun IntArray.getOrNull(index: Int): Int? {
    return if (index >= 0 && index <= lastIndex) get(index) else null
}

/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
@kotlin.internal.InlineOnly
public inline fun LongArray.getOrNull(index: Int): Long? {
    return if (index >= 0 && index <= lastIndex) get(index) else null
}

/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
@kotlin.internal.InlineOnly
public inline fun FloatArray.getOrNull(index: Int): Float? {
    return if (index >= 0 && index <= lastIndex) get(index) else null
}

/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
@kotlin.internal.InlineOnly
public inline fun DoubleArray.getOrNull(index: Int): Double?
public fun IntArray.indexOfFirst(predicate: (Int) -> Boolean): Int {
    for (index in indices) {
        if (predicate(this[index])) {
            return index
        }
    }
    return -1
}

/**
 * Returns index of the first element matching the
 * given [predicate], or -1 if the array does not contain
 * such element.
 */
public fun <@kotlin.internal.OnlyInputTypes T> Array<out T>.indexOfFirst(predicate: (T) -> Boolean): Int {
    if (element == null) {
        for (index in indices) {
            if (element == this[index]) {
                return index
            }
        }
    } else {
        for (index in indices) {
            if (element == this[index]) {
                return index
            }
        }
    } else {
        for (index in indices) {
            if (element == this[index]) {
                return index
            }
        }
    } else {
        for (index in indices) {
            if (element == this[index]) {
                return index
            }
        }
    } else {
        for (index in indices) {
            if (element == this[index]) {
                return index
            }
        }
    } else {
        for (index in indices) {
            if (element == this[index]) {
                return index
            }
        }
    } else {
        for (index in indices) {
            if (element == this[index]) {
                return index
            }
        }
    } else {
        for (index in indices) {
            if (element == this[index]) {
                return index
            }
        }
    } else {
        for (index in indices) {
            if (element == this[index]) {
                return index
            }
        }
    } else {
        for (index in indices) {
            if (element == this[index]) {
                return index
            }
        }
    } else {
        for (index in indices) {
            if (element == this[index]) {
                return index
            }
        }
    } else {
        for (index in indices) {
            if (element == this[index]) {
                return index
            }
        }
    } else {
        for (index in indices) {
            if (element == this[index]) {
                return index
            }
        }
    } else {
        for (index in indices) {
            if (element == this[index]) {
                return index
            }
        }
    } else {
        for (index in indices) {
            if (element == this[index]) {
                return index
            }
        }
    } else {
        for (index in indices) {
            if (element == this[index]) {
                return index
            }
        }
    } else {
        for (index in indices) {
            if (element == this[index]) {
                return index
            }
        }
    } else {
        for (index in indices) {
            if (element == this[index]) {
                return index
            }
        }
    } else {
        for (index in indices) {
            if (element == this[index]) {
given [predicate], or -1 if the array does not contain such element.

LongArray.indexOfLast(predicate: (Long) -> Boolean): Int {\n   for (index in indices) {\n      if
      (predicate(this[index])) {\n         return index\n      }\n   }\n   return -1\n}/**
 * Returns index of the first element matching the given [predicate], or -1 if the array does not contain such element.
 */

FloatArray.indexOfLast(predicate: (Float) -> Boolean): Int {\n   for (index in indices) {\n      if
      (predicate(this[index])) {\n         return index\n      }\n   }\n   return -1\n}/**
 * Returns index of the first element matching the given [predicate], or -1 if the array does not contain such element.
 */

DoubleArray.indexOfLast(predicate: (Double) -> Boolean): Int {\n   for (index in indices) {\n      if
      (predicate(this[index])) {\n         return index\n      }\n   }\n   return -1\n}/**
 * Returns index of the first element matching the given [predicate], or -1 if the array does not contain such element.
 */

BooleanArray.indexOfLast(predicate: (Boolean) -> Boolean): Int {\n   for (index in indices) {\n      if
      (predicate(this[index])) {\n         return index\n      }\n   }\n   return -1\n}/**
 * Returns index of the first element matching the given [predicate], or -1 if the array does not contain such element.
 */

CharArray.indexOfLast(predicate: (Char) -> Boolean): Int {\n   for (index in indices.reversed()) {\n      if
      (predicate(this[index])) {\n         return index\n      }\n   }\n   return -1\n}/**
 * Returns index of the last element matching the given [predicate], or -1 if the array does not contain such element.
 */

BooleanArray.indexOfFirst(predicate: (Boolean) -> Boolean): Int {\n   for (index in indices) {\n      if
      (predicate(this[index])) {\n         return index\n      }\n   }\n   return -1\n}/**
 * Returns index of the first element matching the given [predicate], or -1 if the array does not contain such element.
 */

CharArray.indexOfFirst(predicate: (Char) -> Boolean): Int {\n   for (index in indices) {\n      if
      (predicate(this[index])) {\n         return index\n      }\n   }\n   return -1\n}/**
 * Returns index of the first element matching the given [predicate], or -1 if the array does not contain such element.
 */

ByteBuffer.indexOfLast(predicate: (Byte) -> Boolean): Int {\n   for (index in indices.reversed()) {\n      if
      (predicate(this[index])) {\n         return index\n      }\n   }\n   return -1\n}/**
 * Returns index of the last element matching the given [predicate], or -1 if the array does not contain such element.
 */

CharArray.indexOfFirst(predicate: (Char) -> Boolean): Int {\n   for (index in indices) {\n      if
      (predicate(this[index])) {\n         return index\n      }\n   }\n   return -1\n}/**
 * Returns index of the first element matching the given [predicate], or -1 if the array does not contain such element.
 */

ByteBuffer.indexOfFirst(predicate: (Byte) -> Boolean): Int {\n   for (index in indices) {\n      if
      (predicate(this[index])) {\n         return index\n      }\n   }\n   return -1\n}/**
 * Returns index of the first element matching the given [predicate], or -1 if the array does not contain such element.
 */

DoubleBuffer.indexOfLast(predicate: (Double) -> Boolean): Int {\n   for (index in indices.reversed()) {\n      if
      (predicate(this[index])) {\n         return index\n      }\n   }\n   return -1\n}/**
 * Returns index of the last element matching the given [predicate], or -1 if the array does not contain such element.
 */

DoubleBuffer.indexOfFirst(predicate: (Double) -> Boolean): Int {\n   for (index in indices) {\n      if
      (predicate(this[index])) {\n         return index\n      }\n   }\n   return -1\n}/**
 * Returns index of the first element matching the given [predicate], or -1 if the array does not contain such element.
 */

IntArray.indexOfLast(predicate: (Int) -> Boolean): Int {\n   for (index in indices.reversed()) {\n      if
      (predicate(this[index])) {\n         return index\n      }\n   }\n   return -1\n}/**
 * Returns index of the last element matching the given [predicate], or -1 if the array does not contain such element.
 */

IntArray.indexOfFirst(predicate: (Int) -> Boolean): Int {\n   for (index in indices) {\n      if
      (predicate(this[index])) {\n         return index\n      }\n   }\n   return -1\n}/**
 * Returns index of the first element matching the given [predicate], or -1 if the array does not contain such element.
 */

IntBuffer.indexOfLast(predicate: (Int) -> Boolean): Int {\n   for (index in indices.reversed()) {\n      if
      (predicate(this[index])) {\n         return index\n      }\n   }\n   return -1\n}/**
 * Returns index of the last element matching the given [predicate], or -1 if the array does not contain such element.
 */

IntBuffer.indexOfFirst(predicate: (Int) -> Boolean): Int {\n   for (index in indices) {\n      if
      (predicate(this[index])) {\n         return index\n      }\n   }\n   return -1\n}/**
 * Returns index of the first element matching the given [predicate], or -1 if the array does not contain such element.
 */

ByteArray.indexOfLast(predicate: (Byte) -> Boolean): Int {\n   for (index in indices.reversed()) {\n      if
      (predicate(this[index])) {\n         return index\n      }\n   }\n   return -1\n}/**
 * Returns index of the last element matching the given [predicate], or -1 if the array does not contain such element.
 */

ByteArray.indexOfFirst(predicate: (Byte) -> Boolean): Int {\n   for (index in indices) {\n      if
      (predicate(this[index])) {\n         return index\n      }\n   }\n   return -1\n}/**
 * Returns index of the first element matching the given [predicate], or -1 if the array does not contain such element.
 */

ShortBuffer.indexOfLast(predicate: (Short) -> Boolean): Int {\n   for (index in indices.reversed()) {\n      if
      (predicate(this[index])) {\n         return index\n      }\n   }\n   return -1\n}/**
 * Returns index of the last element matching the given [predicate], or -1 if the array does not contain such element.
 */

ShortBuffer.indexOfFirst(predicate: (Short) -> Boolean): Int {\n   for (index in indices) {\n      if
      (predicate(this[index])) {\n         return index\n      }\n   }\n   return -1\n}/**
 * Returns index of the first element matching the given [predicate], or -1 if the array does not contain such element.
 */

DoubleArray.indexOfLast(predicate: (Double) -> Boolean): Int {\n   for (index in indices.reversed()) {\n      if
      (predicate(this[index])) {\n         return index\n      }\n   }\n   return -1\n}/**
 * Returns index of the last element matching the given [predicate], or -1 if the array does not contain such element.
 */

DoubleArray.indexOfFirst(predicate: (Double) -> Boolean): Int {\n   for (index in indices) {\n      if
      (predicate(this[index])) {\n         return index\n      }\n   }\n   return -1\n}/**
 * Returns index of the first element matching the given [predicate], or -1 if the array does not contain such element.
 */

FloatArray.indexOfLast(predicate: (Float) -> Boolean): Int {\n   for (index in indices.reversed()) {\n      if
      (predicate(this[index])) {\n         return index\n      }\n   }\n   return -1\n}/**
 * Returns index of the last element matching the given [predicate], or -1 if the array does not contain such element.
 */

FloatArray.indexOfFirst(predicate: (Float) -> Boolean): Int {\n   for (index in indices) {\n      if
      (predicate(this[index])) {\n         return index\n      }\n   }\n   return -1\n}/**
 * Returns index of the first element matching the given [predicate], or -1 if the array does not contain such element.
 */

LongBuffer.indexOfLast(predicate: (Long) -> Boolean): Int {\n   for (index in indices.reversed()) {\n      if
      (predicate(this[index])) {\n         return index\n      }\n   }\n   return -1\n}/**
 * Returns index of the last element matching the given [predicate], or -1 if the array does not contain such element.
 */

LongBuffer.indexOfFirst(predicate: (Long) -> Boolean): Int {\n   for (index in indices) {\n      if
      (predicate(this[index])) {\n         return index\n      }\n   }\n   return -1\n}/**
 * Returns index of the first element matching the given [predicate], or -1 if the array does not contain such element.
 */

LongArray.indexOfLast(predicate: (Long) -> Boolean): Int {\n   for (index in indices.reversed()) {\n      if
      (predicate(this[index])) {\n         return index\n      }\n   }\n   return -1\n}/**
 * Returns index of the last element matching the given [predicate], or -1 if the array does not contain such element.
 */

LongArray.indexOfFirst(predicate: (Long) -> Boolean): Int {\n   for (index in indices) {\n      if
      (predicate(this[index])) {\n         return index\n      }\n   }\n   return -1\n}/**
 * Returns index of the first element matching the given [predicate], or -1 if the array does not contain such element.
 */

samples.collections.Collections.Elements.lastIndexOf(<T> Array<out T>:Int(predicate: (T) -> Boolean): Int {\n   for (index in indices) {\n      if
      (predicate(this[index])) {\n         return index\n      }\n   }\n   return -1\n}/**
 * Returns index of the first element matching the given [predicate], or -1 if the array does not contain such element.
 */

samples.collections.Collections.Elements.indexOfFirst(<T> Array<out T>:Int(predicate: (T) -> Boolean): Int {\n   for (index in indices) {\n      if
      (predicate(this[index])) {\n         return index\n      }\n   }\n   return -1\n}/**
 * Returns index of the first element matching the given [predicate], or -1 if the array does not contain such element.
 */

samples.collections.Collections.Elements.lastIndexOf(ByteArray last): Byte {\n   if (isEmpty())
```
element.

```java
* @throws NoSuchElementException if the array is empty.

public fun ShortArray.last(): Short {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[lastIndex]
}

/**
 * Returns the last element.
 * 
 * @throws NoSuchElementException if the array is empty.
 * 
 * @sample samples.collections.Collections.Elements.last
 */
public fun IntArray.last(): Int {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[lastIndex]
}

/**
 * Returns the last element.
 * 
 * @throws NoSuchElementException if the array is empty.
 * 
 * @sample samples.collections.Collections.Elements.last
 */
public fun LongArray.last(): Long {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[lastIndex]
}

/**
 * Returns the last element.
 * 
 * @throws NoSuchElementException if the array is empty.
 * 
 * @sample samples.collections.Collections.Elements.last
 */
public fun FloatArray.last(): Float {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[lastIndex]
}

/**
 * Returns the last element.
 * 
 * @throws NoSuchElementException if the array is empty.
 * 
 * @sample samples.collections.Collections.Elements.last
 */
public fun DoubleArray.last(): Double {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[lastIndex]
}

/**
 * Returns the last element.
 * 
 * @throws NoSuchElementException if the array is empty.
 * 
 * @sample samples.collections.Collections.Elements.last
 */
public fun BooleanArray.last(): Boolean {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[lastIndex]
}

/**
 * Returns the last element.
 * 
 * @throws NoSuchElementException if the array is empty.
 * 
 * @sample samples.collections.Collections.Elements.last
 */
public fun CharArray.last(): Char {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[lastIndex]
}
```

```java
* @throws NoSuchElementException if no such element is found.

public inline fun <T> Array<out T>.last(predicate: (T) -> Boolean): T {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns the last element matching the given [predicate].
 * 
 * @throws NoSuchElementException if no such element is found.
 * 
 * @sample samples.collections.Collections.Elements.last
 */
public inline fun ByteArray.last(predicate: (Byte) -> Boolean): Byte {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns the last element matching the given [predicate].
 * 
 * @throws NoSuchElementException if no such element is found.
 * 
 * @sample samples.collections.Collections.Elements.last
 */
public inline fun ShortArray.last(predicate: (Short) -> Boolean): Short {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns the last element matching the given [predicate].
 * 
 * @throws NoSuchElementException if no such element is found.
 * 
 * @sample samples.collections.Collections.Elements.last
 */
public inline fun IntArray.last(predicate: (Int) -> Boolean): Int {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns the last element matching the given [predicate].
 * 
 * @throws NoSuchElementException if no such element is found.
 * 
 * @sample samples.collections.Collections.Elements.last
 */
public inline fun LongArray.last(predicate: (Long) -> Boolean): Long {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns the last element matching the given [predicate].
 * 
 * @throws NoSuchElementException if no such element is found.
 * 
 * @sample samples.collections.Collections.Elements.last
 */
public inline fun FloatArray.last(predicate: (Float) -> Boolean): Float {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    throw NoSuchElementException("Array contains no element matching the predicate.")
}
```
Float {\n  for (index in this.indices.reversed()) {\n    val element = this[index]\n    if (predicate(element)) return element\n  }\n  throw NoSuchElementException("Array contains no element matching the predicate.")\n} \n\n/**\n * Returns the last element matching the given \[predicate\].\n * \n * @throws NoSuchElementException if no such element is found.\n * \n * @sample samples.collections.Collections.Elements.lastIndexOf\n */\npublic inline fun DoubleArray.lastIndexOf(predicate: (Double) -> Boolean): Double {\n  for (index in this.indices.reversed()) {\n    val element = this[index]\n    if (predicate(element)) return element\n  }\n  throw NoSuchElementException("Array contains no element matching the predicate.")\n} \n\n/**\n * Returns the last element matching the given \[predicate\].\n * \n * @throws NoSuchElementException if no such element is found.\n * \n * @sample samples.collections.Collections.Elements.lastIndexOf\n */\npublic inline fun BooleanArray.lastIndexOf(predicate: (Boolean) -> Boolean): Boolean {\n  for (index in this.indices.reversed()) {\n    val element = this[index]\n    if (predicate(element)) return element\n  }\n  throw NoSuchElementException("Array contains no element matching the predicate.")\n} \n\n/**\n * Returns the last element matching the given \[predicate\].\n * \n * @throws NoSuchElementException if no such element is found.\n * \n * @sample samples.collections.Collections.Elements.lastIndexOf\n */\npublic inline fun CharArray.lastIndexOf(predicate: (Char) -> Boolean): Char {\n  for (index in this.indices.reversed()) {\n    val element = this[index]\n    if (predicate(element)) return element\n  }\n  throw NoSuchElementException("Array contains no element matching the predicate.")\n} \n\n/**\n * Returns last index of \[element\], or -1 if the array does not contain element.\n * \n * @sample samples.collections.Collections.Elements.lastIndexOf\n */\npublic <@kotlin.internal.OnlyInputTypes T> Array<out T>.lastIndexOf(element: T): Int {\n  for (index in indices.reversed()) {\n    if (this[index] == null) {\n      return index\n    }\n  }\n  return -1\n}\n\n/**\n * Returns last index of \[element\], or -1 if the array does not contain element.\n * \n * @sample samples.collections.Collections.Elements.lastIndexOf\n */\npublic fun ByteArray.lastIndexOf(element: Byte): Int {\n  for (index in indices.reversed()) {\n    if (element == this[index]) {\n      return index\n    }\n  }\n  return -1\n}\n\n/**\n * Returns last index of \[element\], or -1 if the array does not contain element.\n * \n * @sample samples.collections.Collections.Elements.lastIndexOf\n */\npublic fun ShortArray.lastIndexOf(element: Short): Int {\n  for (index in indices.reversed()) {\n    if (element == this[index]) {\n      return index\n    }\n  }\n  return -1\n}\n\n/**\n * Returns last index of \[element\], or -1 if the array does not contain element.\n * \n * @sample samples.collections.Collections.Elements.lastIndexOf\n */\npublic fun IntArray.lastIndexOf(element: Int): Int {\n  for (index in indices.reversed()) {\n    if (element == this[index]) {\n      return index\n    }\n  }\n  return -1\n}\n\n/**\n * Returns last index of \[element\], or -1 if the array does not contain element.\n * \n * @sample samples.collections.Collections.Elements.lastIndexOf\n */\npublic fun LongArray.lastIndexOf(element: Long): Int {\n  for (index in indices.reversed()) {\n    if (element == this[index]) {\n      return index\n    }\n  }\n  return -1\n}\n\n/**\n * Returns last index of \[element\], or -1 if the array does not contain element.\n * \n * @sample samples.collections.Collections.Elements.lastIndexOf\n */\npublic fun FloatArray.lastIndexOf(element: Float): Int {\n  for (index in indices.reversed()) {\n    if (element == this[index]) {\n      return index\n    }\n  }\n  return -1\n}\n\n/**\n * Returns last index of \[element\], or -1 if the array does not contain element.\n * \n * @sample samples.collections.Collections.Elements.lastIndexOf\n */\npublic fun DoubleArray.lastIndexOf(element: Double): Int {\n  for (index in indices.reversed()) {\n    if (element == this[index]) {\n      return index\n    }\n  }\n  return -1\n}\n\n/**\n * Returns last index of \[element\], or -1 if the array does not contain element.\n * \n * @sample samples.collections.Collections.Elements.lastIndexOf\n */\npublic fun BooleanArray.lastIndexOf(element: Boolean): Int {\n  for (index in indices.reversed()) {\n    if (element == this[index]) {\n      return index\n    }\n  }\n  return -1\n}\n\n/**\n * Returns last index of \[element\], or -1 if the array does not contain element.\n * \n * @sample samples.collections.Collections.Elements.lastIndexOf\n */\npublic fun CharArray.lastIndexOf(element: Char): Int {\n  for (index in indices.reversed()) {\n    if (element == this[index]) {\n      return index\n    }\n  }\n  return -1\n}
empty.

```kotlin
public fun <T> Array<out T>.lastOrNull(): T? {
    return if (isEmpty()) null else this[size - 1]
}

public fun ByteArray.lastOrNull(): Byte? {
    return if (isEmpty()) null else this[size - 1]
}

public fun ShortArray.lastOrNull(): Short? {
    return if (isEmpty()) null else this[size - 1]
}

public fun IntArray.lastOrNull(): Int? {
    return if (isEmpty()) null else this[size - 1]
}

public fun LongArray.lastOrNull(): Long? {
    return if (isEmpty()) null else this[size - 1]
}

public fun FloatArray.lastOrNull(): Float? {
    return if (isEmpty()) null else this[size - 1]
}

public fun BooleanArray.lastOrNull(): Boolean? {
    return if (isEmpty()) null else this[size - 1]
}

public fun CharArray.lastOrNull(): Char? {
    return if (isEmpty()) null else this[size - 1]
}

public inline fun <T> Array<out T>.lastOrNull(predicate: (T) -> Boolean): T? {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    return null
}

public inline fun ByteArray.lastOrNull(predicate: (Byte) -> Boolean): Byte? {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    return null
}

public inline fun ShortArray.lastOrNull(predicate: (Short) -> Boolean): Short? {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    return null
}

public inline fun IntArray.lastOrNull(predicate: (Int) -> Boolean): Int? {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    return null
}

public inline fun LongArray.lastOrNull(predicate: (Long) -> Boolean): Long? {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    return null
}

public inline fun FloatArray.lastOrNull(predicate: (Float) -> Boolean): Float? {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    return null
}

public inline fun DoubleArray.lastOrNull(predicate: (Double) -> Boolean): Double? {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    return null
}
```

Open Source Used In security_automation identity-guest-cs 1186
-> Boolean: Boolean? {\n    for (index in this.indices.reversed()) {\n        val element = this[index]\n        if (predicate(element)) return element\n    }\n    return null\n}\n
/**
 * Returns the last element matching the given [predicate], or `null` if no such element was found.
 * 
 * @param predicate A function that maps elements of the source to booleans.
 * @return The last element that satisfies the [predicate], or `null` if no such element exists.
 * 
 * @sample samples.collections.Collections.Elements.last
 */
public inline fun CharArray.lastOrNull(predicate: (Char) -> Boolean): Char? {
    for (index in this.indices.reversed()) {
        val element = this[index]\n        if (predicate(element)) return element
    }
    return null
}

/**
 * Returns a random element from this array.
 * 
 * @throws NoSuchElementException if this array is empty.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun <T> Array<out T>.random(): T {
    return random(Random())\n}\n
/**
 * Returns a random element from this array using the specified source of randomness.
 * 
 * @throws NoSuchElementException if this array is empty.
 */
@SinceKotlin("1.3")
public fun <T> Array<out T>.random(random: Random): T {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return get(random.nextInt(size))
}

/**
 * Returns a random element from this array using the specified source of randomness.
 * 
 * @throws NoSuchElementException if this array is empty.
 */
@SinceKotlin("1.3")
public fun ByteArray.random(random: Random): Byte {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return get(random.nextInt(size))
}

/**
 * Returns a random element from this array using the specified source of randomness.
 * 
 * @throws NoSuchElementException if this array is empty.
 */
@SinceKotlin("1.3")
public fun ShortArray.random(random: Random): Short {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return get(random.nextInt(size))
}

/**
 * Returns a random element from this array using the specified source of randomness.
 * 
 * @throws NoSuchElementException if this array is empty.
 */
@SinceKotlin("1.3")
public fun IntArray.random(random: Random): Int {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return get(random.nextInt(size))
}

/**
 * Returns a random element from this array using the specified source of randomness.
 * 
 * @throws NoSuchElementException if this array is empty.
 */
@SinceKotlin("1.3")
public fun LongArray.random(random: Random): Long {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return get(random.nextInt(size))
}

/**
 * Returns a random element from this array using the specified source of randomness.
 * 
 * @throws NoSuchElementException if this array is empty.
 */
@SinceKotlin("1.3")
public fun FloatArray.random(random: Random): Float {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return get(random.nextInt(size))
}

/**
 * Returns a random element from this array using the specified source of randomness.
 * 
 * @throws NoSuchElementException if this array is empty.
 */
@SinceKotlin("1.3")
public fun DoubleArray.random(random: Random): Double {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return get(random.nextInt(size))
}

/**
 * Returns a random element from this array using the specified source of randomness.
 * 
 * @throws NoSuchElementException if this array is empty.
 */
@SinceKotlin("1.3")
public fun BooleanArray.random(random: Random): Boolean {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return get(random.nextInt(size))
}

/**
 * Returns a random element from this array using the specified source of randomness.
 * 
 * @throws NoSuchElementException if this array is empty.
 */
@SinceKotlin("1.3")
public fun CharArray.random(random: Random): Char {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return get(random.nextInt(size))
}
package com.example

import java.util.Random

public fun ShortArray.randomOrNull(random: Random): Short? {
    if (isEmpty())
        return null
    return get(random.nextInt(size))
}

/**
 * Returns a random element from this array using the specified source of randomness, or `null` if this array is empty.
 */
public fun IntArray.randomOrNull(random: Random): Int? {
    if (isEmpty())
        return null
    return get(random.nextInt(size))
}

/**
 * Returns a random element from this array using the specified source of randomness, or `null` if this array is empty.
 */
public fun LongArray.randomOrNull(random: Random): Long? {
    if (isEmpty())
        return null
    return get(random.nextInt(size))
}

/**
 * Returns a random element from this array using the specified source of randomness, or `null` if this array is empty.
 */
public funFloatArray.randomOrNull(random: Random): Float? {
    if (isEmpty())
        return null
    return get(random.nextInt(size))
}

/**
 * Returns a random element from this array using the specified source of randomness, or `null` if this array is empty.
 */
public fun DoubleArray.randomOrNull(random: Random): Double? {
    if (isEmpty())
        return null
    return get(random.nextInt(size))
}

/**
 * Returns a random element from this array using the specified source of randomness, or `null` if this array is empty.
 */
public fun BooleanArray.randomOrNull(random: Random): Boolean? {
    if (isEmpty())
        return null
    return get(random.nextInt(size))
}

/**
 * Returns a random element from this array using the specified source of randomness, or `null` if this array is empty.
 */
public fun CharArray.randomOrNull(random: Random): Char? {
    if (isEmpty())
        return null
    return get(random.nextInt(size))
}

/**
 * Returns the single element, or throws an exception if the array is empty or has more than one element.
 */
public fun <T> Array<out T>.single(): T {
    return when (size) {
        0 -> throw NoSuchElementException("Array is empty.")
        1 -> this[0]
        else -> throw IllegalArgumentException("Array has more than one element.")
    }
}

/**
 * Returns the single element, or throws an exception if the array is empty or has more than one element.
 */
public fun ByteArray.single(): Byte {
    return when (size) {
        0 -> throw NoSuchElementException("Array is empty.")
        1 -> this[0]
        else -> throw IllegalArgumentException("Array has more than one element.")
    }
}

/**
 * Returns the single element, or throws an exception if the array is empty or has more than one element.
 */
public fun ShortArray.single(): Short {
    return when (size) {
        0 -> throw NoSuchElementException("Array is empty.")
        1 -> this[0]
        else -> throw IllegalArgumentException("Array has more than one element.")
    }
}

/**
 * Returns the single element, or throws an exception if the array is empty or has more than one element.
 */
public fun IntArray.single(): Int {
    return when (size) {
        0 -> throw NoSuchElementException("Array is empty.")
        1 -> this[0]
        else -> throw IllegalArgumentException("Array has more than one element.")
    }
}

/**
 * Returns the single element, or throws an exception if the array is empty or has more than one element.
 */
public fun LongArray.single(): Long {
    return when (size) {
        0 -> throw NoSuchElementException("Array is empty.")
        1 -> this[0]
        else -> throw IllegalArgumentException("Array has more than one element.")
    }
}

/**
 * Returns the single element, or throws an exception if the array is empty or has more than one element.
 */
public funFloatArray.single(): Float {
    return when (size) {
        0 -> throw NoSuchElementException("Array is empty.")
        1 -> this[0]
        else -> throw IllegalArgumentException("Array has more than one element.")
    }
}

/**
 * Returns the single element, or throws an exception if the array is empty or has more than one element.
 */
public fun DoubleArray.single(): Double {
    return when (size) {
        0 -> throw NoSuchElementException("Array is empty.")
        1 -> this[0]
        else -> throw IllegalArgumentException("Array has more than one element.")
    }
}
NoSuchElementException("Array is empty.")
1 -> this[0]
else -> throw
IllegalArgumentException("Array has more than one element.")

public fun BooleanArray.single(): Boolean {
    return when (size) {
        0 -> throw NoSuchElementException("Array is empty.")
        1 -> this[0]
        else -> throw IllegalArgumentException("Array has more than one element.")
    }
}

/**
 * Returns the single element, or throws an exception if the array is empty or has more than one element.
 */
public fun CharArray.single(): Char {
    return when (size) {
        0 -> throw NoSuchElementException("Array is empty.")
        1 -> this[0]
        else -> throw IllegalArgumentException("Array has more than one element.")
    }
}

/**
 * Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element.
 */
public inline fun <T> Array<out T>.single(predicate: (T) -> Boolean): T {
    var single: T? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found)
                throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found)
        throw NoSuchElementException("Array contains no element matching the predicate.")
    return single as T
}

/**
 * Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element.
 */
public inline fun ByteArray.single(predicate: (Byte) -> Boolean): Byte {
    var single: Byte? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found)
                throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found)
        throw NoSuchElementException("Array contains no element matching the predicate.")
    return single as Byte
}

/**
 * Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element.
 */
public inline fun ShortArray.single(predicate: (Short) -> Boolean): Short {
    var single: Short? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found)
                throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found)
        throw NoSuchElementException("Array contains no element matching the predicate.")
    return single as Short
}

/**
 * Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element.
 */
public inline fun IntArray.single(predicate: (Int) -> Boolean): Int {
    var single: Int? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found)
                throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found)
        throw NoSuchElementException("Array contains no element matching the predicate.")
    return single as Int
}

/**
 * Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element.
 */
public inline fun LongArray.single(predicate: (Long) -> Boolean): Long {
    var single: Long? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found)
                throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found)
        throw NoSuchElementException("Array contains no element matching the predicate.")
    return single as Long
}

/**
 * Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element.
 */
public inline fun FloatArray.single(predicate: (Float) -> Boolean): Float {
    var single: Float? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found)
                throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found)
        throw NoSuchElementException("Array contains no element matching the predicate.")
    return single as Float
}

/**
 * Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element.
 */
public inline fun DoubleArray.single(predicate: (Double) -> Boolean): Double {
    var single: Double? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found)
                throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found)
        throw NoSuchElementException("Array contains no element matching the predicate.")
    return single as Double
}
Extensions to the `Array` class for finding single elements:

```kotlin
/**
 * Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element.
 */
@Suppress("UNCHECKED_CAST")
public inline fun <T> Array<out T>.singleOrNull(predicate: (T) -> Boolean): T? {
    var single: T? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.

            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no matching element.

    @Suppress("UNCHECKED_CAST")
    return single as T
}

/**
 * Returns single element, or `null` if the array is empty or has more than one element.
 */
@Suppress("UNCHECKED_CAST")
public fun CharArray.singleOrNull(): Char? {
    return if (size == 1) this[0] else null
}

/**
 * Returns single element, or `null` if the array is empty or has more than one element.
 */
public fun BooleanArray.singleOrNull(): Boolean {
    var single: Boolean? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.

            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no matching element.

    @Suppress("UNCHECKED_CAST")
    return single as Boolean
}

/**
 * Returns single element, or `null` if the array is empty or has more than one element.
 */
public fun ByteArray.singleOrNull(): Byte? {
    return if (size == 1) this[0] else null
}

/**
 * Returns single element, or `null` if the array is empty or has more than one element.
 */
public fun DoubleArray.singleOrNull(): Double? {
    return if (size == 1) this[0] else null
}

/**
 * Returns single element, or `null` if the array is empty or has more than one element.
 */
public fun FloatArray.singleOrNull(): Float? {
    return if (size == 1) this[0] else null
}

/**
 * Returns single element, or `null` if the array is empty or has more than one element.
 */
public fun IntArray.singleOrNull(): Int? {
    return if (size == 1) this[0] else null
}

/**
 * Returns single element, or `null` if the array is empty or has more than one element.
 */
public fun ShortArray.singleOrNull(): Short? {
    return if (size == 1) this[0] else null
}

/**
 * Returns single element, or `null` if the array is empty or has more than one element.
 */
public fun CharArray.single(predicate: (Char) -> Boolean): Char
```
single: Long? = null
var found = false
for (element in this) { 
  if (predicate(element)) { 
    if (found) return null
    single = element
    found = true
  } 
  if (!found) return null
}

single: Float? = null
var found = false
for (element in this) { 
  if (predicate(element)) { 
    if (found) return null
    single = element
    found = true
  } 
  if (!found) return null
}

single: Double? = null
var found = false
for (element in this) { 
  if (predicate(element)) { 
    if (found) return null
    single = element
    found = true
  } 
  if (!found) return null
}

single: Char? = null
var found = false
for (element in this) { 
  if (predicate(element)) { 
    if (found) return null
    single = element
    found = true
  } 
  if (!found) return null
}

samples.collections.Collections.Transformations.drop(n: Int) { 
  require(n >= 0) { 
    "Requested element count \$n is less than zero." 
  } 
  return takeLast((size - n).coerceAtLeast(0)).toArray() 
}

samples.collections.Collections.Transformations.drop(n: Int) { 
  require(n >= 0) { 
    "Requested element count \$n is less than zero." 
  } 
  return takeLast((size - n).coerceAtLeast(0)).toByteArray() 
}

samples.collections.Collections.Transformations.drop(n: Int) { 
  require(n >= 0) { 
    "Requested element count \$n is less than zero." 
  } 
  return takeLast((size - n).coerceAtLeast(0)).toShortArray() 
}

samples.collections.Collections.Transformations.drop(n: Int) { 
  require(n >= 0) { 
    "Requested element count \$n is less than zero." 
  } 
  return takeLast((size - n).coerceAtLeast(0)).toIntArray() 
}

samples.collections.Collections.Transformations.drop(n: Int) { 
  require(n >= 0) { 
    "Requested element count \$n is less than zero." 
  } 
  return takeLast((size - n).coerceAtLeast(0)).toCharArray() 
}

samples.collections.Collections.Transformations.drop(n: Int) { 
  require(n >= 0) { 
    "Requested element count \$n is less than zero." 
  } 
  return takeLast((size - n).coerceAtLeast(0)).toList() 
}
IllegalArgumentException if \([n]\) is negative.\n * \n * @throws samples.collections.Collections.Transformations.drop\n */
public fun BooleanArray.dropLast(n: Int): List<Boolean> {
    require(n >= 0) { "Requested element count \([n]\) is less than zero." }
    return takeLast((size - n).coerceAtLeast(0)).\n        \n        * Returns a list containing all elements except last \([n]\) elements.\n * \n * @throws samples.collections.Collections.Transformations.drop\n */
public fun CharArray.dropLast(n: Int): List<Char> {
    require(n >= 0) { "Requested element count \([n]\) is less than zero." }
    return takeLast((size - n).coerceAtLeast(0)).\n        \n        * Returns a list containing all elements except last \([n]\) elements.\n * \n * @throws samples.collections.Collections.Transformations.drop\n */
public fun <T> Array<out T>.dropLast(n: Int): List<T> {
    require(n >= 0) { "Requested element count \([n]\) is less than zero." }
    return take((size - n).coerceAtLeast(0)).\n        \n        * Returns a list containing all elements except last \([n]\) elements.\n * \n * @throws samples.collections.Collections.Transformations.drop\n */
public fun ByteArray.dropLast(n: Int): List<Byte> {
    require(n >= 0) { "Requested element count \([n]\) is less than zero." }
    return take((size - n).coerceAtLeast(0)).\n        \n        * Returns a list containing all elements except last \([n]\) elements.\n * \n * @throws samples.collections.Collections.Transformations.drop\n */
public fun ShortArray.dropLast(n: Int): List<Short> {
    require(n >= 0) { "Requested element count \([n]\) is less than zero." }
    return take((size - n).coerceAtLeast(0)).\n        \n        * Returns a list containing all elements except last \([n]\) elements.\n * \n * @throws samples.collections.Collections.Transformations.drop\n */
public fun IntArray.dropLast(n: Int): List<Int> {
    require(n >= 0) { "Requested element count \([n]\) is less than zero." }
    return take((size - n).coerceAtLeast(0)).\n        \n        * Returns a list containing all elements except last \([n]\) elements.\n * \n * @throws samples.collections.Collections.Transformations.drop\n */
public fun LongArray.dropLast(n: Int): List<Long> {
    require(n >= 0) { "Requested element count \([n]\) is less than zero." }
    return take((size - n).coerceAtLeast(0)).\n        \n        * Returns a list containing all elements except last \([n]\) elements.\n * \n * @throws samples.collections.Collections.Transformations.drop\n */
public fun FloatArray.dropLast(n: Int): List<Float> {
    require(n >= 0) { "Requested element count \([n]\) is less than zero." }
    return take((size - n).coerceAtLeast(0)).\n        \n        * Returns a list containing all elements except last \([n]\) elements.\n * \n * @throws samples.collections.Collections.Transformations.drop\n */
public fun DoubleArray.dropLast(n: Int): List<Double> {
    require(n >= 0) { "Requested element count \([n]\) is less than zero." }
    return take((size - n).coerceAtLeast(0)).\n        \n        * Returns a list containing all elements except last \([n]\) elements.\n * \n * @throws samples.collections.Collections.Transformations.drop\n */
public fun BooleanArray.dropLastWhile(predicate: (T) -> Boolean): List<T> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return take(index + 1)
        }
    }
    return emptyList()
}

/**
 * Returns a list containing all elements except last elements that satisfy the given 
 * \[predicate\].\n * \n * @throws samples.collections.Collections.Transformations.drop\n */
public fun <T> Array<out T>.dropLastWhile(predicate: (T) -> Boolean): List<T> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return take(index + 1)
        }
    }
    return emptyList()
}

/**
 * Returns a list containing all elements except last elements that satisfy the given 
 * \[predicate\].\n * \n * @throws samples.collections.Collections.Transformations.drop\n */
public fun <T> Array<out T>.dropLastWhile(predicate: (T) -> Boolean): List<T> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return take(index + 1)
        }
    }
    return emptyList()
}

/**
 * Returns a list containing all elements except last elements that satisfy the given 
 * \[predicate\].\n * \n * @throws samples.collections.Collections.Transformations.drop\n */
public fun <T> Array<out T>.dropLastWhile(predicate: (T) -> Boolean): List<T> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return take(index + 1)
        }
    }
    return emptyList()
}

/**
 * Returns a list containing all elements except last elements that satisfy the given 
 * \[predicate\].\n * \n * @throws samples.collections.Collections.Transformations.drop\n */
public fun <T> Array<out T>.dropLastWhile(predicate: (T) -> Boolean): List<T> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return take(index + 1)
        }
    }
    return emptyList()
}

/**
 * Returns a list containing all elements except last elements that satisfy the given 
 * \[predicate\].\n * \n * @throws samples.collections.Collections.Transformations.drop\n */
public fun <T> Array<out T>.dropLastWhile(predicate: (T) -> Boolean): List<T> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return take(index + 1)
        }
    }
    return emptyList()
}

/**
 * Returns a list containing all elements except last elements that satisfy the given 
 * \[predicate\].\n * \n * @throws samples.collections.Collections.Transformations.drop\n */
public fun <T> Array<out T>.dropLastWhile(predicate: (T) -> Boolean): List<T> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return take(index + 1)
        }
    }
    return emptyList()
}

/**
 * Returns a list containing all elements except last elements that satisfy the given 
 * \[predicate\].\n * \n * @throws samples.collections.Collections.Transformations.drop\n */
public fun <T> Array<out T>.dropLastWhile(predicate: (T) -> Boolean): List<T> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return take(index + 1)
        }
    }
    return emptyList()
samples.collections.Collections.Transformations.drop\n\n/*public inline fun ByteArray.dropLastWhile(predicate: (Byte) -> Boolean): List<Byte> {\n    for (index in lastIndex downTo 0) {\n        if (!predicate(this[index])) {\n            return \n        }\n    }\n    return emptyList()\n}\n*/

public inline fun ByteArray.dropLastWhile(predicate: (Byte) -> Boolean): List<Byte> {\n    for (index in lastIndex downTo 0) {\n        if (!predicate(this[index])) {\n            return take(index + 1)\n        }\n    }\n    return emptyList()\n}

/**
 * Returns a list containing all elements except last elements that satisfy the given [predicate].\n *\n * @sample samples.collections.Collections.Transformations.drop\n */
public inline fun ShortArray.dropLastWhile(predicate: (Short) -> Boolean): List<Short> {\n    for (index in lastIndex downTo 0) {\n        if (!predicate(this[index])) {\n            return take(index + 1)\n        }\n    }\n    return emptyList()\n}

/**
 * Returns a list containing all elements except last elements that satisfy the given [predicate].\n *\n * @sample samples.collections.Collections.Transformations.drop\n */
public inline fun IntArray.dropLastWhile(predicate: (Int) -> Boolean): List<Int> {\n    for (index in lastIndex downTo 0) {\n        if (!predicate(this[index])) {\n            return take(index + 1)\n        }\n    }\n    return emptyList()\n}

/**
 * Returns a list containing all elements except last elements that satisfy the given [predicate].\n *\n * @sample samples.collections.Collections.Transformations.drop\n */
public inline fun LongArray.dropLastWhile(predicate: (Long) -> Boolean): List<Long> {\n    for (index in lastIndex downTo 0) {\n        if (!predicate(this[index])) {\n            return take(index + 1)\n        }\n    }\n    return emptyList()\n}

/**
 * Returns a list containing all elements except last elements that satisfy the given [predicate].\n *\n * @sample samples.collections.Collections.Transformations.drop\n */
public inline fun FloatArray.dropLastWhile(predicate: (Float) -> Boolean): List<Float> {\n    for (index in lastIndex downTo 0) {\n        if (!predicate(this[index])) {\n            return take(index + 1)\n        }\n    }\n    return emptyList()\n}

/**
 * Returns a list containing all elements except last elements that satisfy the given [predicate].\n *\n * @sample samples.collections.Collections.Transformations.drop\n */
public inline fun DoubleArray.dropLastWhile(predicate: (Double) -> Boolean): List<Double> {\n    for (index in lastIndex downTo 0) {\n        if (!predicate(this[index])) {\n            return take(index + 1)\n        }\n    }\n    return emptyList()\n}

/**
 * Returns a list containing all elements except last elements that satisfy the given [predicate].\n *\n * @sample samples.collections.Collections.Transformations.drop\n */
public inline fun BooleanArray.dropLastWhile(predicate: (Boolean) -> Boolean): List<Boolean> {\n    for (index in lastIndex downTo 0) {\n        if (!predicate(this[index])) {\n            return take(index + 1)\n        }\n    }\n    return emptyList()\n}

/**
 * Returns a list containing all elements except last elements that satisfy the given [predicate].\n *\n * @sample samples.collections.Collections.Transformations.drop\n */
public inline fun CharArray.dropLastWhile(predicate: (Char) -> Boolean): List<Char> {\n    for (index in lastIndex downTo 0) {\n        if (!predicate(this[index])) {\n            return take(index + 1)\n        }\n    }\n    return emptyList()\n}

/**
 * Returns a list containing all elements except first elements that satisfy the given [predicate].\n *\n * @sample samples.collections.Collections.Transformations.drop\n */
public inline fun <T> Array<out T>.dropWhile(predicate: (T) -> Boolean): List<T> {\n    var yielding = false\n    val list = ArrayList<T>()\n    for (item in this) {\n        if (yielding) {\n            list.add(item)\n        } else if (!predicate(item)) {\n            list.add(item)\n            yielding = true\n        }\n    }\n    return list\n}

/**
 * Returns a list containing all elements except first elements that satisfy the given [predicate].\n *\n * @sample samples.collections.Collections.Transformations.drop\n */
public inline fun ByteArray.dropWhile(predicate: (Byte) -> Boolean): List<Byte> {\n    var yielding = false\n    val list = ArrayList<Byte>()\n    for (item in this) {\n        if (yielding) {\n            list.add(item)\n        } else if (!predicate(item)) {\n            list.add(item)\n            yielding = true\n        }\n    }\n    return list\n}

/**
 * Returns a list containing all elements except first elements that satisfy the given [predicate].\n *\n * @sample samples.collections.Collections.Transformations.drop\n */
public inline fun ShortArray.dropWhile(predicate: (Short) -> Boolean): List<Short> {\n    var yielding = false\n    val list = ArrayList<Short>()\n    for (item in this) {\n        if (yielding) {\n            list.add(item)\n        } else if (!predicate(item)) {\n            list.add(item)\n            yielding = true\n        }\n    }\n    return list\n}

/**
 * Returns a list containing all elements except first elements that satisfy the given [predicate].\n *\n * @sample samples.collections.Collections.Transformations.drop\n */
public inline fun IntArray.dropWhile(predicate: (Int) -> Boolean): List<Int> {\n    var yielding = false\n    val list = ArrayList<Int>()\n    for (item in this) {\n        if (yielding) {\n            list.add(item)\n        } else if (!predicate(item)) {\n            list.add(item)\n            yielding = true\n        }\n    }\n    return list\n}

/**
 * Returns a list containing all elements except first elements that satisfy the given [predicate].\n *\n * @sample samples.collections.Collections.Transformations.drop\n */
public inline fun DoubleArray.dropWhile(predicate: (Double) -> Boolean): List<Double> {\n    var yielding = false\n    val list = ArrayList<Double>()\n    for (item in this) {\n        if (yielding) {\n            list.add(item)\n        } else if (!predicate(item)) {\n            list.add(item)\n            yielding = true\n        }\n    }\n    return list\n}

/**
 * Returns a list containing all elements except first elements that satisfy the given [predicate].\n *\n * @sample samples.collections.Collections.Transformations.drop\n */
public inline fun BooleanArray.dropWhile(predicate: (Boolean) -> Boolean): List<Boolean> {\n    var yielding = false\n    val list = ArrayList<Boolean>() \n    for (item in this) {\n        if (yielding) {\n            list.add(item)\n        } else if (!predicate(item)) {\n            list.add(item)\n            yielding = true\n        }\n    }\n    return list\n}

/**
 * Returns a list containing all elements except first elements that satisfy the given [predicate].\n *\n * @sample samples.collections.Collections.Transformations.drop\n */
public inline fun CharArray.dropWhile(predicate: (Char) -> Boolean): List<Char> {\n    var yielding = false\n    val list = ArrayList<Char>()\n    for (item in this) {\n        if (yielding) {\n            list.add(item)\n        } else if (!predicate(item)) {\n            list.add(item)\n            yielding = true\n        }\n    }\n    return list\n}

/**
 * Returns a list containing all elements except first elements that satisfy the given [predicate].\n *\n * @sample samples.collections.Collections.Transformations.drop\n */
public inline fun <T> Array<out T>.dropWhile(predicate: (T) -> Boolean): List<T> {\n    var yielding = false\n    val list = ArrayList<T>()\n    for (item in this) {\n        if (yielding) {\n            list.add(item)\n        } else if (!predicate(item)) {\n            list.add(item)\n            yielding = true\n        }\n    }\n    return list\n}
list.add(item)
yielding = true
}
return list
}

/**
 * Returns a list containing all elements except first elements that satisfy the
 * given [predicate].
 */
public inline fun LongArray.dropWhile(predicate: (Long) -> Boolean): List<Long> {
    var yielding = false
    val list = ArrayList<Long>()
    for (item in this)
        if (yielding)
            list.add(item)
        else if (!predicate(item)) {
            list.add(item)
            yielding = true
        }
    return list
}

/**
 * Returns a list containing all elements except first elements that satisfy the
 * given [predicate].
 */
public inline fun FloatArray.dropWhile(predicate: (Float) -> Boolean): List<Float> {
    var yielding = false
    val list = ArrayList<Float>()
    for (item in this)
        if (yielding)
            list.add(item)
        else if (!predicate(item)) {
            list.add(item)
            yielding = true
        }
    return list
}

/**
 * Returns a list containing all elements except first elements that satisfy the
 * given [predicate].
 */
public inline fun DoubleArray.dropWhile(predicate: (Double) -> Boolean): List<Double> {
    var yielding = false
    val list = ArrayList<Double>()
    for (item in this)
        if (yielding)
            list.add(item)
        else if (!predicate(item)) {
            list.add(item)
            yielding = true
        }
    return list
}

/**
 * Returns a list containing all elements except first elements that satisfy the
 * given [predicate].
 */
public inline fun BooleanArray.dropWhile(predicate: (Boolean) -> Boolean): List<Boolean> {
    var yielding = false
    val list = ArrayList<Boolean>()
    for (item in this)
        if (yielding)
            list.add(item)
        else if (!predicate(item)) {
            list.add(item)
            yielding = true
        }
    return list
}

/**
 * Returns a list containing all elements except first elements that satisfy the
 * given [predicate].
 */
public inline fun CharArray.dropWhile(predicate: (Char) -> Boolean): List<Char> {
    var yielding = false
    val list = ArrayList<Char>()
    for (item in this)
        if (yielding)
            list.add(item)
        else if (!predicate(item)) {
            list.add(item)
            yielding = true
        }
    return list
}
and the element itself\n* and returns the result of predicate evaluation on the element.\n* \n* @sample samples.collections.Collections.Filtering.filterIndexed\n*/
public inline fun <T> Array<out T>.filterIndexed(predicate: (index: Int, T) -> Boolean): List<T> {
    \n    return filterIndexedTo(ArrayList<T>(), predicate)\n}\n
/**
 * Returns a list containing only elements matching the given \[predicate\].\n* @param \[predicate\] function that takes the index of an element and the element itself\n* and returns the result of predicate evaluation on the element.\n* \n* @sample samples.collections.Collections.Filtering.filterIndexed\n*/
public inline fun ByteArray.filterIndexed(predicate: (index: Int, Byte) -> Boolean): List<Byte> {
    \n    return filterIndexedTo(ArrayList<Byte>(), predicate)\n}\n
/**
 * Returns a list containing only elements matching the given \[predicate\].\n* @param \[predicate\] function that takes the index of an element and the element itself\n* and returns the result of predicate evaluation on the element.\n* \n* @sample samples.collections.Collections.Filtering.filterIndexed\n*/
public inline fun ShortArray.filterIndexed(predicate: (index: Int, Short) -> Boolean): List<Short> {
    \n    return filterIndexedTo(ArrayList<Short>(), predicate)\n}\n
/**
 * Returns a list containing only elements matching the given \[predicate\].\n* @param \[predicate\] function that takes the index of an element and the element itself\n* and returns the result of predicate evaluation on the element.\n* \n* @sample samples.collections.Collections.Filtering.filterIndexed\n*/
public inline fun IntArray.filterIndexed(predicate: (index: Int, Int) -> Boolean): List<Int> {
    \n    return filterIndexedTo(ArrayList<Int>(), predicate)\n}\n
/**
 * Returns a list containing only elements matching the given \[predicate\].\n* @param \[predicate\] function that takes the index of an element and the element itself\n* and returns the result of predicate evaluation on the element.\n* \n* @sample samples.collections.Collections.Filtering.filterIndexed\n*/
public inline fun LongArray.filterIndexed(predicate: (index: Int, Long) -> Boolean): List<Long> {
    \n    return filterIndexedTo(ArrayList<Long>(), predicate)\n}\n
/**
 * Returns a list containing only elements matching the given \[predicate\].\n* @param \[predicate\] function that takes the index of an element and the element itself\n* and returns the result of predicate evaluation on the element.\n* \n* @sample samples.collections.Collections.Filtering.filterIndexed\n*/
public inline fun FloatArray.filterIndexed(predicate: (index: Int, Float) -> Boolean): List<Float> {
    \n    return filterIndexedTo(ArrayList<Float>(), predicate)\n}\n
/**
 * Returns a list containing only elements matching the given \[predicate\].\n* @param \[predicate\] function that takes the index of an element and the element itself\n* and returns the result of predicate evaluation on the element.\n* \n* @sample samples.collections.Collections.Filtering.filterIndexed\n*/
public inline fun DoubleArray.filterIndexed(predicate: (index: Int, Double) -> Boolean): List<Double> {
    \n    return filterIndexedTo(ArrayList<Double>(), predicate)\n}\n
/**
 * Returns a list containing only elements matching the given \[predicate\].\n* @param \[predicate\] function that takes the index of an element and the element itself\n* and returns the result of predicate evaluation on the element.\n* \n* @sample samples.collections.Collections.Filtering.filterIndexed\n*/
public inline fun BooleanArray.filterIndexed(predicate: (index: Int, Boolean) -> Boolean): List<Boolean> {
    \n    return filterIndexedTo(ArrayList<Boolean>(), predicate)\n}\n
/**
 * Returns a list containing only elements matching the given \[predicate\].\n* @param \[predicate\] function that takes the index of an element and the element itself\n* and returns the result of predicate evaluation on the element.\n* \n* @sample samples.collections.Collections.Filtering.filterIndexed\n*/
public inline fun CharArray.filterIndexed(predicate: (index: Int, Char) -> Boolean): List<Char> {
    \n    return filterIndexedTo(ArrayList<Char>(), predicate)\n}\n
/**
* Appends all elements matching the given \[predicate\] to the given \[destination\].\n* \n* @sample samples.collections.Collections.Filtering.filterIndexedTo\n*/
public inline fun <T, C : MutableCollection<in T>> Array<out T>.filterIndexedTo(destination: C, predicate: (index: Int, T) -> Boolean): C {
    forEachIndexed { index, element ->
        if (predicate(index, element)) destination.add(element)
    }
    return destination
}\n
/**
* Appends all elements matching the given \[predicate\] to the given \[destination\].\n* \n* @sample samples.collections.Collections.Filtering.filterIndexedTo\n*/
public inline fun <C : MutableCollection<in Byte>> ByteArray.filterIndexedTo(destination: C, predicate: (index: Int, Byte) -> Boolean): C {
    forEachIndexed { index, element ->
        if (predicate(index, element)) destination.add(element)
    }
    return destination
}
the given [predicate].\n * \n * @sample samples.collections.Collections.Filtering.filter\n
/**
 * Returns a list containing all elements not matching the given [predicate].\n * @sample samples.collections.Collections.Filtering.filter\n */
public inline fun <T> Array<out T>.filterNot(predicate: (T) -> Boolean): List<T> {
    return filterNotTo(ArrayList<T>(), predicate)\n}

/**
 * Returns a list containing all elements not matching the given [predicate].\n * @sample samples.collections.Collections.Filtering.filter\n */
public inline fun ByteArray.filterNot(predicate: (Byte) -> Boolean): List<Byte> {
    return filterNotTo(ArrayList<Byte>(), predicate)\n}

/**
 * Returns a list containing all elements not matching the given [predicate].\n * @sample samples.collections.Collections.Filtering.filter\n */
public inline fun ShortArray.filterNot(predicate: (Short) -> Boolean): List<Short> {
    return filterNotTo(ArrayList<Short>(), predicate)\n}

/**
 * Returns a list containing all elements not matching the given [predicate].\n * @sample samples.collections.Collections.Filtering.filter\n */
public inline fun IntArray.filterNot(predicate: (Int) -> Boolean): List<Int> {
    return filterNotTo(ArrayList<Int>(), predicate)\n}

/**
 * Returns a list containing all elements not matching the given [predicate].\n * @sample samples.collections.Collections.Filtering.filter\n */
public inline fun LongArray.filterNot(predicate: (Long) -> Boolean): List<Long> {
    return filterNotTo(ArrayList<Long>(), predicate)\n}

/**
 * Returns a list containing all elements not matching the given [predicate].\n * @sample samples.collections.Collections.Filtering.filter\n */
public inline fun FloatArray.filterNot(predicate: (Float) -> Boolean): List<Float> {
    return filterNotTo(ArrayList<Float>(), predicate)\n}

/**
 * Returns a list containing all elements not matching the given [predicate].\n * @sample samples.collections.Collections.Filtering.filter\n */
public inline fun DoubleArray.filterNot(predicate: (Double) -> Boolean): List<Double> {
    return filterNotTo(ArrayList<Double>(), predicate)\n}

/**
 * Returns a list containing all elements not matching the given [predicate].\n * @sample samples.collections.Collections.Filtering.filter\n */
public inline fun BooleanArray.filterNot(predicate: (Boolean) -> Boolean): List<Boolean> {
    return filterNotTo(ArrayList<Boolean>(), predicate)\n}

/**
 * Returns a list containing all elements not matching the given [predicate].\n * @sample samples.collections.Collections.Filtering.filter\n */
public inline fun CharArray.filterNot(predicate: (Char) -> Boolean): List<Char> {
    return filterNotTo(ArrayList<Char>(), predicate)\n}

fun <T : Any> Array<out T>.filterNotNull(): List<T> {
    return filterNotNullTo(ArrayList<T>())\n}

fun <C : MutableCollection<in T>, T : Any> Array<out T>.filterNotNullTo(destination: C): C {
    return filterNotNullTo(destination, ArrayList<T>())\n}

fun <T : Any> Array<out T>.filterNotNull(): List<T> {
    return filterNotNullTo(ArrayList<T>())\n}

fun <C : MutableCollection<in T>> Array<out T>.filterNotNullTo(destination: C): C {
    return filterNotNullTo(destination, ArrayList<T>())\n}
LongArray.filterNotTo(destination: C, predicate: (Long) -> Boolean): C

קוד ומשלים מהUMMY: (predicate(element)) destination.add(element) \n
return destination

* Append all elements not matching the given [predicate] to the given [destination].

* @sample

samples.collections.Collections.Filtering.filterTo

//public inline fun <C : MutableCollection<in Float>>

FloatArray.filterNotTo(destination: C, predicate: (Float) -> Boolean): C

* Append all elements not matching the given [predicate] to the given [destination].

* @sample

samples.collections.Collections.Filtering.filterTo

//public inline fun <C : MutableCollection<in Double>>

DoubleArray.filterNotTo(destination: C, predicate: (Double) -> Boolean): C

* Append all elements not matching the given [predicate] to the given [destination].

* @sample

samples.collections.Collections.Filtering.filterTo

//public inline fun <C : MutableCollection<in Boolean>>

BooleanArray.filterNotTo(destination: C, predicate: (Boolean) -> Boolean): C

* Append all elements not matching the given [predicate] to the given [destination].

* @sample

samples.collections.Collections.Filtering.filterTo

//public inline fun <C : MutableCollection<in Char>>

CharArray.filterNotTo(destination: C, predicate: (Char) -> Boolean): C

* Append all elements not matching the given [predicate] to the given [destination].

* @sample

samples.collections.Collections.Filtering.filterTo

//public inline fun <C : MutableCollection<in Byte>>

ByteArray.filterTo(destination: C, predicate: (Byte) -> Boolean): C

* Append all elements matching the given [predicate] to the given [destination].

* @sample

samples.collections.Collections.Filtering.filterTo

//public inline fun <C : MutableCollection<in Short>>

ShortArray.filterTo(destination: C, predicate: (Short) -> Boolean): C

* Append all elements matching the given [predicate] to the given [destination].

* @sample

samples.collections.Collections.Filtering.filterTo

//public inline fun <C : MutableCollection<in Int>>

IntArray.filterTo(destination: C, predicate: (Int) -> Boolean): C

* Append all elements matching the given [predicate] to the given [destination].

* @sample

samples.collections.Collections.Filtering.filterTo

//public inline fun <C : MutableCollection<in Long>>

LongArray.filterTo(destination: C, predicate: (Long) -> Boolean): C

* Append all elements matching the given [predicate] to the given [destination].

* @sample

samples.collections.Collections.Filtering.filterTo

//public inline fun <C : MutableCollection<in Float>>

FloatArray.filterTo(destination: C, predicate: (Float) -> Boolean): C

* Append all elements matching the given [predicate] to the given [destination].

* @sample

samples.collections.Collections.Filtering.filterTo

//public inline fun <C : MutableCollection<in Double>>

DoubleArray.filterTo(destination: C, predicate: (Double) -> Boolean): C

* Append all elements matching the given [predicate] to the given [destination].

* @sample

samples.collections.Collections.Filtering.filterTo

//public inline fun <C : MutableCollection<in Boolean>>

BooleanArray.filterTo(destination: C, predicate: (Boolean) -> Boolean): C

* Append all elements matching the given [predicate] to the given [destination].

* @sample

samples.collections.Collections.Filtering.filterTo

//public inline fun <C : MutableCollection<in Char>>

CharArray.filterTo(destination: C, predicate: (Char) -> Boolean): C

* Append all elements matching the given [predicate] to the given [destination].

* @sample

samples.collections.Collections.Filtering.filterTo
Returns an array containing elements of this array at specified [indices].

public fun DoubleArray.sliceArray(indices: IntRange): DoubleArray {
    if (indices.isEmpty()) return DoubleArray(0)
    return copyOfRange(indices.start, indices.endInclusive + 1)
}

/**
 * Returns an array containing elements of this array at specified [indices].
 */
public fun FloatArray.sliceArray(indices: IntRange): FloatArray {
    if (indices.isEmpty()) return FloatArray(0)
    return copyOfRange(indices.start, indices.endInclusive + 1)
}

/**
 * Returns an array containing elements of this array at specified [indices].
 */
public fun IntArray.sliceArray(indices: IntRange): IntArray {
    if (indices.isEmpty()) return IntArray(0)
    return copyOfRange(indices.start, indices.endInclusive + 1)
}

/**
 * Returns an array containing elements of this array at specified [indices].
 */
public fun LongArray.sliceArray(indices: IntRange): LongArray {
    if (indices.isEmpty()) return LongArray(0)
    return copyOfRange(indices.start, indices.endInclusive + 1)
}

/**
 * Returns an array containing elements of this array at specified [indices].
 */
public fun ShortArray.sliceArray(indices: IntRange): ShortArray {
    if (indices.isEmpty()) return ShortArray(0)
    return copyOfRange(indices.start, indices.endInclusive + 1)
}

/**
 * Returns an array containing elements of this array at specified [indices].
 */
public fun ByteArray.sliceArray(indices: IntRange): ByteArray {
    if (indices.isEmpty()) return ByteArray(0)
    return copyOfRange(indices.start, indices.endInclusive + 1)
}

/**
 * Returns an array containing elements of this array at specified [indices].
 */
public fun <T> Array<T>.sliceArray(indices: Collection<Int>): Array<T> {
    val size = indices.size
    val list = ArrayList<T>(size)
    for (index in indices) {
        list.add(get(index))
    }
    return list
}

/**
 * Returns a list containing elements at specified [indices].
 */
public fun CharArray.slice(indices: Iterable<Int>): List<Char> {
    val size = indices.collectionSizeOrDefault(10)
    if (size == 0) return emptyList()
    val list = ArrayList<Char>(size)
    for (index in indices) {
        list.add(get(index))
    }
    return list
}

@TargetApi(21)
public fun <T> Array<T>.sliceArray(indices: Collection<Int>): Array<T> {

    val result = Array<Array<T>>(indices.size) as Array<Array<T>>
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}

Array<T>.sliceArray(indices: Collection<Int>): Array<T> {

    val result = Array<Array<T>>(indices.size) as Array<Array<T>>
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[targetIndex++]
    }
    return result
}
CharArray.take(n: Int): List<Char> {
    require(n >= 0) { "Requested element count $n is less than zero."
    if (n >= size) return toList()!
    if (n == 1) return listOf(this[0]!) var count = 0
    val list = ArrayList<T>(n) for (item in this) {
        list.add(item) if (++count == n) break
    } return list
}

/**
 * Returns a list containing first [n] elements.
 * 
 * @throws IllegalArgumentException if [n] is negative.
 * @sample samples.collections.Collections.Transformations.take
 */
public fun <T> Array<out T>.take(n: Int): List<T> {
    require(n >= 0) { "Requested element count $n is less than zero."
    if (n >= size) return toList()!
    if (n == 1) return listOf(this[0]!) var count = 0
    val list = ArrayList<T>(n) for (item in this) {
        list.add(item) if (++count == n) break
    } return list
}
Returns a list containing last elements satisfying the given [predicate].

```kotlin
public inline fun <T> takeLastWhile(predicate: (T) -> Boolean): List<T> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return drop(index + 1)
        }
    }
    return toList()
}
```

```kotlin
public inline fun ByteArray.takeLastWhile(predicate: (Byte) -> Boolean): List<Byte> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return drop(index + 1)
        }
    }
    return toList()
}
```

```kotlin
public inline fun ShortArray.takeLastWhile(predicate: (Short) -> Boolean): List<Short> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return drop(index + 1)
        }
    }
    return toList()
}
```

```kotlin
public inline fun IntArray.takeLastWhile(predicate: (Int) -> Boolean): List<Int> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return drop(index + 1)
        }
    }
    return toList()
}
```

```kotlin
public inline fun LongArray.takeLastWhile(predicate: (Long) -> Boolean): List<Long> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return drop(index + 1)
        }
    }
    return toList()
}
```

```kotlin
public inline fun FloatArray.takeLastWhile(predicate: (Float) -> Boolean): List<Float> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return drop(index + 1)
        }
    }
    return toList()
}
```

```kotlin
public inline fun DoubleArray.takeLastWhile(predicate: (Double) -> Boolean): List<Double> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return drop(index + 1)
        }
    }
    return toList()
}
```

```kotlin
public inline fun BooleanArray.takeLastWhile(predicate: (Boolean) -> Boolean): List<Boolean> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return drop(index + 1)
        }
    }
    return toList()
}
```

```kotlin
public inline fun CharArray.takeLastWhile(predicate: (Char) -> Boolean): List<Char> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return drop(index + 1)
        }
    }
    return toList()
}
```

```kotlin
public inline fun <T> takeWhile(predicate: (T) -> Boolean): List<T> {
    val list = ArrayList<T>()
    for (item in this) {
        if (!predicate(item))
            break
        list.add(item)
    }
    return list
}
```

```kotlin
public inline fun ByteArray.takeWhile(predicate: (Byte) -> Boolean): List<Byte> {
    val list = ArrayList<Byte>()
    for (item in this) {
        if (!predicate(item))
            break
        list.add(item)
    }
    return list
}
```

```kotlin
public inline fun ShortArray.takeWhile(predicate: (Short) -> Boolean): List<Short> {
    val list = ArrayList<Short>()
    for (item in this) {
        if (!predicate(item))
            break
        list.add(item)
    }
    return list
}
```

```kotlin
public inline fun IntArray.takeWhile(predicate: (Int) -> Boolean): List<Int> {
    val list = ArrayList<Int>()
    for (item in this) {
        if (!predicate(item))
            break
        list.add(item)
    }
    return list
}
```

```kotlin
public inline fun LongArray.takeWhile(predicate: (Long) -> Boolean): List<Long> {
    val list = ArrayList<Long>()
    for (item in this) {
        if (!predicate(item))
            break
        list.add(item)
    }
    return list
}
```

```kotlin
public inline fun FloatArray.takeWhile(predicate: (Float) -> Boolean): List<Float> {
    val list = ArrayList<Float>()
    for (item in this) {
        if (!predicate(item))
            break
        list.add(item)
    }
    return list
}
```

```kotlin
public inline fun DoubleArray.takeWhile(predicate: (Double) -> Boolean): List<Double> {
    val list = ArrayList<Double>()
    for (item in this) {
        if (!predicate(item))
            break
        list.add(item)
    }
    return list
}
```

```kotlin
public inline fun BooleanArray.takeWhile(predicate: (Boolean) -> Boolean): List<Boolean> {
    val list = ArrayList<Boolean>()
    for (item in this) {
        if (!predicate(item))
            break
        list.add(item)
    }
    return list
}
```

```kotlin
public inline fun CharArray.takeWhile(predicate: (Char) -> Boolean): List<Char> {
    val list = ArrayList<Char>()
    for (item in this) {
        if (!predicate(item))
            break
        list.add(item)
    }
    return list
}
```
Array<T>.reverse(fromIndex: Int, toIndex: Int): Unit { 
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    val midPoint = (fromIndex + toIndex) / 2
    if (fromIndex == midPoint) return
    var reverseIndex = 
    toIndex - 1 
    for (index in fromIndex until midPoint) { 
        val tmp = this[index]
        this[index] = 
        this[reverseIndex]
        this[reverseIndex] = tmp
        reverseIndex--
    }
}

/**
 * Reverses elements of the 
array in the specified range in-place.
 * 
 * @param fromIndex the start of the range (inclusive) to reverse.
 * @param toIndex the end of the range (exclusive) to reverse.
 * 
 * @throws IndexOutOfBoundsException if 
[fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
@SinceKotlin("1.4")
public fun ByteArray.reverse(fromIndex: Int, toIndex: Int): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    val midPoint = (fromIndex + toIndex) / 2
    if (fromIndex == midPoint) return
    var reverseIndex = 
    toIndex - 1 
    for (index in fromIndex until midPoint) { 
        val tmp = this[index]
        this[index] = 
        this[reverseIndex]
        this[reverseIndex] = tmp
        reverseIndex--
    }
}

/**
 * Reverses elements of the 
array in the specified range in-place.
 * 
 * @param fromIndex the start of the range (inclusive) to reverse.
 * @param toIndex the end of the range (exclusive) to reverse.
 * 
 * @throws IndexOutOfBoundsException if 
[fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
@SinceKotlin("1.4")
public fun ShortArray.reverse(fromIndex: Int, toIndex: Int): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    val midPoint = (fromIndex + toIndex) / 2
    if (fromIndex == midPoint) return
    var reverseIndex = 
    toIndex - 1 
    for (index in fromIndex until midPoint) { 
        val tmp = this[index]
        this[index] = 
        this[reverseIndex]
        this[reverseIndex] = tmp
        reverseIndex--
    }
}

/**
 * Reverses elements of the 
array in the specified range in-place.
 * 
 * @param fromIndex the start of the range (inclusive) to reverse.
 * @param toIndex the end of the range (exclusive) to reverse.
 * 
 * @throws IndexOutOfBoundsException if 
[fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
@SinceKotlin("1.4")
public fun IntArray.reverse(fromIndex: Int, toIndex: Int): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    val midPoint = (fromIndex + toIndex) / 2
    if (fromIndex == midPoint) return
    var reverseIndex = 
    toIndex - 1 
    for (index in fromIndex until midPoint) { 
        val tmp = this[index]
        this[index] = 
        this[reverseIndex]
        this[reverseIndex] = tmp
        reverseIndex--
    }
}

/**
 * Reverses elements of the 
array in the specified range in-place.
 * 
 * @param fromIndex the start of the range (inclusive) to reverse.
 * @param toIndex the end of the range (exclusive) to reverse.
 * 
 * @throws IndexOutOfBoundsException if 
[fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
@SinceKotlin("1.4")
public fun LongArray.reverse(fromIndex: Int, toIndex: Int): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    val midPoint = (fromIndex + toIndex) / 2
    if (fromIndex == midPoint) return
    var reverseIndex = 
    toIndex - 1 
    for (index in fromIndex until midPoint) { 
        val tmp = this[index]
        this[index] = 
        this[reverseIndex]
        this[reverseIndex] = tmp
        reverseIndex--
    }
}

/**
 * Reverses elements of the 
array in the specified range in-place.
 * 
 * @param fromIndex the start of the range (inclusive) to reverse.
 * @param toIndex the end of the range (exclusive) to reverse.
 * 
 * @throws IndexOutOfBoundsException if 
[fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
@SinceKotlin("1.4")
public fun FloatArray.reverse(fromIndex: Int, toIndex: Int): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    val midPoint = (fromIndex + toIndex) / 2
    if (fromIndex == midPoint) return
    var reverseIndex = 
    toIndex - 1 
    for (index in fromIndex until midPoint) { 
        val tmp = this[index]
        this[index] = 
        this[reverseIndex]
        this[reverseIndex] = tmp
        reverseIndex--
    }
}
DoubleArray.reverse(fromIndex: Int, toIndex: Int): Unit {\n    \n    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n    val midPoint = (fromIndex + toIndex) / 2\n    if (fromIndex == midPoint) return\n    var reverseIndex = toIndex - 1\n    for (index in fromIndex until midPoint) {\n        val tmp = this[index]\n        this[index] = this[reverseIndex]\n        this[reverseIndex] = tmp\n        reverseIndex--\n    }\n}\n
/**\n * Reverses elements of the array in the specified range in-place.\n * @param fromIndex the end of the range (exclusive) to reverse.\n * @param toIndex the start of the range (inclusive) to reverse.\n */\n@SinceKotlin("1.4")@Throws(IndexOutOfBoundsException\n    @Throws(\n        IllegalArgumentException\n    )\n    \n    public fun BooleanArray.reverse(fromIndex: Int, toIndex: Int): Unit {\n        \n        AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n        val midPoint = (fromIndex + toIndex) / 2\n        if (fromIndex == midPoint) return\n        var reverseIndex = toIndex - 1\n        for (index in fromIndex until midPoint) {\n            val tmp = this[index]\n            this[index] = this[reverseIndex]\n            this[reverseIndex] = tmp\n            reverseIndex--\n        }\n    }

/**\n * Reverses elements of the array in the specified range in-place.\n */\n@SinceKotlin("1.4")@Throws(IndexOutOfBoundsException\n    @Throws(\n        IllegalArgumentException\n    )\n    \n    public fun CharArray.reverse(fromIndex: Int, toIndex: Int): Unit {\n        \n        AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n        val midPoint = (fromIndex + toIndex) / 2\n        if (fromIndex == midPoint) return\n        var reverseIndex = toIndex - 1\n        for (index in fromIndex until midPoint) {\n            val tmp = this[index]\n            this[index] = this[reverseIndex]\n            this[reverseIndex] = tmp\n            reverseIndex--\n        }\n    }

/**\n * Returns a list with elements in reversed order.\n */\npublic fun <T> Array<out T>.reversed(): List<T> {\n    if (isEmpty()) return emptyList()\n    val list = toMutableList()\n    list.reverse()\n    return list\n}

/**\n * Returns a list with elements in reversed order.\n */\npublic fun ByteArray.reversed(): List<Byte> {\n    if (isEmpty()) return emptyList()\n    val list = toMutableList()\n    list.reverse()\n    return list\n}

/**\n * Returns a list with elements in reversed order.\n */\npublic fun ShortArray.reversed(): List<Short> {\n    if (isEmpty()) return emptyList()\n    val list = toMutableList()\n    list.reverse()\n    return list\n}

/**\n * Returns a list with elements in reversed order.\n */\npublic fun IntArray.reversed(): List<Int> {\n    if (isEmpty()) return emptyList()\n    val list = toMutableList()\n    list.reverse()\n    return list\n}

/**\n * Returns a list with elements in reversed order.\n */\npublic fun LongArray.reversed(): List<Long> {\n    if (isEmpty()) return emptyList()\n    val list = toMutableList()\n    list.reverse()\n    return list\n}

/**\n * Returns a list with elements in reversed order.\n */\npublic fun FloatArray.reversed(): List<Float> {\n    if (isEmpty()) return emptyList()\n    val list = toMutableList()\n    list.reverse()\n    return list\n}

/**\n * Returns a list with elements in reversed order.\n */\npublic fun DoubleArray.reversed(): List<Double> {\n    if (isEmpty()) return emptyList()\n    val list = toMutableList()\n    list.reverse()\n    return list\n}

/**\n * Returns a list with elements in reversed order.\n */\npublic fun BooleanArray.reversed(): List<Boolean> {\n    if (isEmpty()) return emptyList()\n    val list = toMutableList()\n    list.reverse()\n    return list\n}

/**\n * Returns a list with elements in reversed order.\n */\npublic fun CharArray.reversed(): List<Char> {\n    if (isEmpty()) return emptyList()\n    val list = toMutableList()\n    list.reverse()\n    return list\n}

/**\n * Returns an array with elements of this array in reversed order.\n */\n@SinceKotlin("1.4")@Throws(IndexOutOfBoundsException\n    @Throws(\n        IllegalArgumentException\n    )\n    \n    public fun byteArrayReversedArray(): ByteArray {\n        \n        val lastIndex = size - 1\n        var result = ByteArray(size)\n        for (i in 0..lastIndex) {\n            result[lastIndex - i] = this[i]\n        }\n        return result\n    }

/**\n * Returns an array with elements of this array in reversed order.\n */\n@SinceKotlin("1.4")@Throws(IndexOutOfBoundsException\n    @Throws(\n        IllegalArgumentException\n    )\n    \n    public fun shortArrayReversedArray(): ShortArray {\n        \n        val lastIndex = size - 1\n        var result = ShortArray(size)\n        for (i in 0..lastIndex) {\n            result[lastIndex - i] = this[i]\n        }\n        return result\n    }

/**\n * Returns an array with elements of this array in reversed order.\n */\n@SinceKotlin("1.4")@Throws(IndexOutOfBoundsException\n    @Throws(\n        IllegalArgumentException\n    )\n    \n    public fun charArrayReversedArray(): CharArray {\n        \n        val lastIndex = size - 1\n        var result = CharArray(size)\n        for (i in 0..lastIndex) {\n            result[lastIndex - i] = this[i]\n        }\n        return result\n    }

/**\n * Returns an array with elements of this array in reversed order.\n */\n@SinceKotlin("1.4")@Throws(IndexOutOfBoundsException\n    @Throws(\n        IllegalArgumentException\n    )\n    \n    public fun intArrayReversedArray(): IntArray {\n        \n        val lastIndex = size - 1\n        var result = IntArray(size)\n        for (i in 0..lastIndex) {\n            result[lastIndex - i] = this[i]\n        }\n        return result\n    }
return result

public fun LongArray.reversedArray(): LongArray {
    if (isEmpty()) return this
    val lastIndex = lastIndex
    for (i in 0..lastIndex)
        result[lastIndex - i] = this[i]
    return result
}

/**
 * Returns an array with elements of this array in reversed order.
 */
public fun DoubleArray.reversedArray(): DoubleArray {
    if (isEmpty()) return this
    val result = DoubleArray(size)
    val lastIndex = lastIndex
    for (i in 0..lastIndex)
        result[lastIndex - i] = this[i]
    return result
}

/**
 * Returns an array with elements of this array in reversed order.
 */
public fun BooleanArray.reversedArray(): BooleanArray {
    if (isEmpty()) return this
    val result = BooleanArray(size)
    val lastIndex = lastIndex
    for (i in 0..lastIndex)
        result[lastIndex - i] = this[i]
    return result
}

/**
 * Returns an array with elements of this array in reversed order.
 */
public fun CharArray.reversedArray(): CharArray {
    if (isEmpty()) return this
    val result = CharArray(size)
    val lastIndex = lastIndex
    for (i in 0..lastIndex)
        result[lastIndex - i] = this[i]
    return result
}

/**
 * Randomly shuffles elements in this array in-place.
 */
@SinceKotlin("1.4")
public fun <T> Array<T>.shuffle(): Unit {
    shuffle(Random)
}

/**
 * Randomly shuffles elements in this array in-place using the specified [random] instance as the
 * source of randomness.
 * 
 * See: https://en.wikipedia.org/wiki/Fisher%E2%80%93Yates_shuffle#The_modern_algorithm
 */
@SinceKotlin("1.4")
public fun ByteArray.shuffle(random: Random): Unit {
    shuffle(random)
}

/**
 * Randomly shuffles elements in this array in-place using the specified [random] instance as the
 * source of randomness.
 * 
 * See: https://en.wikipedia.org/wiki/Fisher%E2%80%93Yates_shuffle#The_modern_algorithm
 */
@SinceKotlin("1.4")
public fun ShortArray.shuffle(random: Random): Unit {
    shuffle(random)
}

/**
 * Randomly shuffles elements in this array in-place using the specified [random] instance as the
 * source of randomness.
 * 
 * See: https://en.wikipedia.org/wiki/Fisher%E2%80%93Yates_shuffle#The_modern_algorithm
 */
@SinceKotlin("1.4")
public fun IntArray.shuffle(random: Random): Unit {
    shuffle(random)
}

/**
 * Randomly shuffles elements in this array in-place using the specified [random] instance as the
 * source of randomness.
 * 
 * See: https://en.wikipedia.org/wiki/Fisher%E2%80%93Yates_shuffle#The_modern_algorithm
 */
@SinceKotlin("1.4")
public fun LongArray.shuffle(random: Random): Unit {
    shuffle(random)
}

/**
 * Randomly shuffles elements in this array in-place using the specified [random] instance as the
 * source of randomness.
 * 
 * See: https://en.wikipedia.org/wiki/Fisher%E2%80%93Yates_shuffle#The_modern_algorithm
 */
@SinceKotlin("1.4")
public fun FloatArray.shuffle(random: Random): Unit {
    shuffle(random)
}

/**
 * Randomly shuffles elements in this array in-place using the specified [random] instance as the
 * source of randomness.
 * 
 * See: https://en.wikipedia.org/wiki/Fisher%E2%80%93Yates_shuffle#The_modern_algorithm
 */
@SinceKotlin("1.4")
public fun DoubleArray.shuffle(random: Random): Unit {
    shuffle(random)
}

/**
 * Randomly shuffles elements in this array in-place using the specified [random] instance as the
 * source of randomness.
 * 
 * See: https://en.wikipedia.org/wiki/Fisher%E2%80%93Yates_shuffle#The_modern_algorithm
 */
@SinceKotlin("1.4")
public fun BooleanArray.shuffle(random: Random): Unit {
    shuffle(random)
}

/**
 * Randomly shuffles elements in this array in-place using the specified [random] instance as the
 * source of randomness.
 * 
 * See: https://en.wikipedia.org/wiki/Fisher%E2%80%93Yates_shuffle#The_modern_algorithm
 */
@SinceKotlin("1.4")
public fun CharArray.shuffle(random: Random): Unit {
    shuffle(random)
}
source of randomness.

See:
https://en.wikipedia.org/wiki/Fisher%E2%80%93Yates_shuffle#The_modern_algorithm

Randomly shuffles elements in this array in-place using the specified [random] instance as the source of randomness.

See:
https://en.wikipedia.org/wiki/Fisher%E2%80%93Yates_shuffle#The_modern_algorithm

Randomly shuffles elements in this array in-place using the specified [random] instance as the source of randomness.

See:
https://en.wikipedia.org/wiki/Fisher%E2%80%93Yates_shuffle#The_modern_algorithm

Randomly shuffles elements in this array in-place using the specified [random] instance as the source of randomness.

See:
https://en.wikipedia.org/wiki/Fisher%E2%80%93Yates_shuffle#The_modern_algorithm

Randomly shuffles elements in this array in-place using the specified [random] instance as the source of randomness.

See:
https://en.wikipedia.org/wiki/Fisher%E2%80%93Yates_shuffle#The_modern_algorithm

Randomly shuffles elements in this array in-place using the specified [random] instance as the source of randomness.

See:
https://en.wikipedia.org/wiki/Fisher%E2%80%93Yates_shuffle#The_modern_algorithm

Randomly shuffles elements in this array in-place using the specified [random] instance as the source of randomness.

See:
https://en.wikipedia.org/wiki/Fisher%E2%80%93Yates_shuffle#The_modern_algorithm

Randomly shuffles elements in this array in-place using the specified [random] instance as the source of randomness.

See:
https://en.wikipedia.org/wiki/Fisher%E2%80%93Yates_shuffle#The_modern_algorithm

Randomly shuffles elements in this array in-place using the specified [random] instance as the source of randomness.

See:
https://en.wikipedia.org/wiki/Fisher%E2%80%93Yates_shuffle#The_modern_algorithm

Randomly shuffles elements in this array in-place using the specified [random] instance as the source of randomness.

See:
https://en.wikipedia.org/wiki/Fisher%E2%80%93Yates_shuffle#The_modern_algorithm

Randomly shuffles elements in this array in-place using the specified [random] instance as the source of randomness.

See:
https://en.wikipedia.org/wiki/Fisher%E2%80%93Yates_shuffle#The_modern_algorithm

Randomly shuffles elements in this array in-place using the specified [random] instance as the source of randomness.

See:
https://en.wikipedia.org/wiki/Fisher%E2%80%93Yates_shuffle#The_modern_algorithm

Randomly shuffles elements in this array in-place using the specified [random] instance as the source of randomness.

See:
https://en.wikipedia.org/wiki/Fisher%E2%80%93Yates_shuffle#The_modern_algorithm

Randomly shuffles elements in this array in-place using the specified [random] instance as the source of randomness.

See:
https://en.wikipedia.org/wiki/Fisher%E2%80%93Yates_shuffle#The_modern_algorithm

Randomly shuffles elements in this array in-place using the specified [random] instance as the source of randomness.

See:
https://en.wikipedia.org/wiki/Fisher%E2%80%93Yates_shuffle#The_modern_algorithm

Randomly shuffles elements in this array in-place using the specified [random] instance as the source of randomness.

See:
https://en.wikipedia.org/wiki/Fisher%E2%80%93Yates_shuffle#The_modern_algorithm

Randomly shuffles elements in this array in-place using the specified [random] instance as the source of randomness.

See:
public fun CharArray.sortedArrayDescending(): CharArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sortDescending() }
}

/**
 * Returns an array with all elements of this array sorted descending according to their natural sort order.
 * The sort is \_stable\_. It means that equal elements preserve their order relative to each other after sorting.
 * @sample samples.collections.Collections.Sorting.sortedBy
 */
public <T, R : Comparable<R>> Array<T>.sortedByDescending(selector: (T) -> R?): List<T> {
    return sortedWith(compareByDescending(selector))
}

/**
 * Returns a list of all elements sorted according to natural sort order of the value returned by specified \[selector\] function.
 * The sort is \_stable\_. It means that equal elements preserve their order relative to each other after sorting.
 * @sample samples.collections.Collections.Sorting.sortedBy
 */
public <T, R : Comparable<R>> Array<T>.sortedByWith(selector: (T) -> R?): List<T> {
    return sortedWith(compareByWith(selector))
}

/**
 * Returns a list of all elements sorted according to the specified \[comparator\] function.
 * The sort is \_stable\_. It means that equal elements preserve their order relative to each other after sorting.
 * @sample samples.collections.Collections.Sorting.sortedBy
 */
public <T> Array<T>.sortedByWith(comparator: Comparator<T>): List<T> {
    return sortedWith(compareWith(comparator))
}

/**
 * Returns an array with all elements of this array sorted according to the specified \[comparator\] function.
 * The sort is \_stable\_. It means that equal elements preserve their order relative to each other after sorting.
 * @sample samples.collections.Collections.Sorting.sortedBy
 */
public <T> CharArray.sortedByWith(comparator: Comparator<T>): List<Char> {
    return sortedWith(compareWith(comparator))
}

/**
 * Returns an array with all elements of this array sorted according to the specified \[comparator\] function.
 * The sort is \_stable\_. It means that equal elements preserve their order relative to each other after sorting.
 * @sample samples.collections.Collections.Sorting.sortedBy
 */
public <T> CharArray.sortedByDescendingWith(comparator: Comparator<T>): List<Char> {
    return sortedWith(compareDescendingWith(comparator))
}
according to natural sort order of the value returned by specified [selector] function.

```kotlin
/**
 * Returns a list of all elements sorted descending according to natural sort order of the value returned by specified [selector] function.
 */
public inline fun <R : Comparable<R>> ByteArray.sortedByDescending(crossinline selector: (Byte) -> R?): List<Byte> {
    return sortedWith(compareByDescending(selector))
}

/**
 * Returns a list of all elements sorted descending according to natural sort order of the value returned by specified [selector] function.
 */
public inline fun <R : Comparable<R>> ShortArray.sortedByDescending(crossinline selector: (Short) -> R?): List<Short> {
    return sortedWith(compareByDescending(selector))
}

/**
 * Returns a list of all elements sorted descending according to natural sort order of the value returned by specified [selector] function.
 */
public inline fun <R : Comparable<R>> IntArray.sortedByDescending(crossinline selector: (Int) -> R?): List<Int> {
    return sortedWith(compareByDescending(selector))
}

/**
 * Returns a list of all elements sorted descending according to natural sort order of the value returned by specified [selector] function.
 */
public inline fun <R : Comparable<R>> LongArray.sortedByDescending(crossinline selector: (Long) -> R?): List<Long> {
    return sortedWith(compareByDescending(selector))
}

/**
 * Returns a list of all elements sorted descending according to natural sort order of the value returned by specified [selector] function.
 */
public inline fun <R : Comparable<R>> FloatArray.sortedByDescending(crossinline selector: (Float) -> R?): List<Float> {
    return sortedWith(compareByDescending(selector))
}

/**
 * Returns a list of all elements sorted descending according to natural sort order of the value returned by specified [selector] function.
 */
public inline fun <R : Comparable<R>> DoubleArray.sortedByDescending(crossinline selector: (Double) -> R?): List<Double> {
    return sortedWith(compareByDescending(selector))
}

/**
 * Returns a list of all elements sorted descending according to natural sort order of the value returned by specified [selector] function.
 */
public inline fun <R : Comparable<R>> BooleanArray.sortedByDescending(crossinline selector: (Boolean) -> R?): List<Boolean> {
    return sortedWith(compareByDescending(selector))
}

/**
 * Returns a list of all elements sorted descending according to their natural sort order.
 * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.
 */
public fun <T : Comparable<T>> Array<out T>.sortedDescending(): List<T> {
    return sortedWith(reverseOrder())
}

/**
 * Returns a list of all elements sorted descending according to their natural sort order.
 */
public fun ByteArray.sortedDescending(): List<Byte> {
    return copyOf().apply { sort() }.reversed()
}

/**
 * Returns a list of all elements sorted descending according to their natural sort order.
 */
public fun ShortArray.sortedDescending(): List<Short> {
    return copyOf().apply { sort() }.reversed()
}

/**
 * Returns a list of all elements sorted descending according to their natural sort order.
 */
public fun IntArray.sortedDescending(): List<Int> {
    return copyOf().apply { sort() }.reversed()
}

/**
 * Returns a list of all elements sorted descending according to their natural sort order.
 */
public fun LongArray.sortedDescending(): List<Long> {
    return copyOf().apply { sort() }.reversed()
}

/**
 * Returns a list of all elements sorted descending according to their natural sort order.
 */
public fun FloatArray.sortedDescending(): List<Float> {
    return copyOf().apply { sort() }.reversed()
}

/**
 * Returns a list of all elements sorted descending according to their natural sort order.
 */
public fun DoubleArray.sortedDescending(): List<Double> {
    return copyOf().apply { sort() }.reversed()
}

/**
 * Returns a list of all elements sorted according to the specified [comparator].
 * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.
 */
public fun <T> Array<out T>.sortedWith(comparator: Comparator<in T>): List<T> {
    return sortedArrayWith(comparator).asList()
}

/**
 * Returns a list of all elements sorted according to the specified [comparator].
 */
public fun ByteArray.sortedWith(comparator: Comparator<in Byte>): List<Byte> {
    return toTypedArray().apply { sortWith(comparator) }.asList()
}

/**
 * Returns a list of all elements sorted according to the specified [comparator].
 */
public fun ShortArray.sortedWith(comparator: Comparator<in Short>): List<Short> {
    return toTypedArray().apply { sortWith(comparator) }.asList()
}
elements sorted according to the specified [comparator].

```kotlin
/*
 * Returns a list of all elements sorted according to the specified [comparator].
 */
public fun IntArray.sortedWith(comparator: Comparator<in Int>): List<Int> {
    return toTypedArray().apply { sortWith(comparator) }.asList()
}

/**
 * Returns a list of all elements sorted according to the specified [comparator].
 */
public fun LongArray.sortedWith(comparator: Comparator<in Long>): List<Long> {
    return toTypedArray().apply { sortWith(comparator) }.asList()
}

/**
 * Returns a list of all elements sorted according to the specified [comparator].
 */
public fun FloatArray.sortedWith(comparator: Comparator<in Float>): List<Float> {
    return toTypedArray().apply { sortWith(comparator) }.asList()
}

/**
 * Returns a list of all elements sorted according to the specified [comparator].
 */
public fun DoubleArray.sortedWith(comparator: Comparator<in Double>): List<Double> {
    return toTypedArray().apply { sortWith(comparator) }.asList()
}

/**
 * Returns a list of all elements sorted according to the specified [comparator].
 */
public fun BooleanArray.sortedWith(comparator: Comparator<in Boolean>): List<Boolean> {
    return toTypedArray().apply { sortWith(comparator) }.asList()
}

/**
 * Returns a list of all elements sorted according to the specified [comparator].
 */
public fun CharArray.sortedWith(comparator: Comparator<in Char>): List<Char> {
    return toTypedArray().apply { sortWith(comparator) }.asList()
}
```

@SinceKotlin("1.1")
@kotlin.internal.LowPriorityInOverloadResolution
public expect fun <T> Array<out T>.contentDeepEquals(other: Array<out T>): Boolean

@SinceKotlin("1.4")
public expect infix fun <T> Array<out T>?.contentDeepEquals(other: Array<out T>?): Boolean

@SinceKotlin("1.1")
@kotlin.internal.LowPriorityInOverloadResolution
public expect fun <T> Array<out T>.contentDeepHashCode(): Int

@SinceKotlin("1.4")
public expect infix fun <T> Array<out T>?.contentDeepHashCode(): Int

@SinceKotlin("1.1")
@kotlin.internal.LowPriorityInOverloadResolution
public expect fun <T> Array<out T>.contentDeepToString(): String

@SinceKotlin("1.4")
public expect infix fun <T> Array<out T>?.contentDeepToString(): String
reference
* is rendered as `\[\ldots\]` to prevent recursion.

```kotlin
expect fun <T> Array<out T>.contentDeepToString(): String

/*
 * Returns `true` if the two specified arrays are
 * structurally* equal to one another.
 * i.e. contain the same number of the same elements in the same order.
 * The elements are compared for equality with the [equals][Any.equals] function.
 * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
 */
@SinceKotlin("1.4")
public expect fun
ByteArray.contentDeepToString(): String

/*
 * Returns `true` if the two specified arrays are
 * structurally* equal to one another.
 * i.e. contain the same number of the same elements in the same order.
 * The elements are compared for equality with the [equals][Any.equals] function.
 * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
 */
@SinceKotlin("1.1")
@DeprecatedSinceKotlin(hiddenSince = "1.4")
public expect infix fun ShortArray.contentEquals(other: ShortArray): Boolean

/*
 * Returns `true` if the two specified arrays are
 * structurally* equal to one another.
 * i.e. contain the same number of the same elements in the same order.
 * The elements are compared for equality with the [equals][Any.equals] function.
 * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
 */
@SinceKotlin("1.1")
@DeprecatedSinceKotlin(hiddenSince = "1.4")
public expect infix fun IntArray.contentEquals(other: IntArray): Boolean

/*
 * Returns `true` if the two specified arrays are
 * structurally* equal to one another.
 * i.e. contain the same number of the same elements in the same order.
 * The elements are compared for equality with the [equals][Any.equals] function.
 * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
 */
@SinceKotlin("1.1")
@DeprecatedSinceKotlin(hiddenSince = "1.4")
public expect infix fun LongArray.contentEquals(other: LongArray): Boolean

/*
 * Returns `true` if the two specified arrays are
 * structurally* equal to one another.
 * i.e. contain the same number of the same elements in the same order.
 * The elements are compared for equality with the [equals][Any.equals] function.
 * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
 */
@SinceKotlin("1.1")
@DeprecatedSinceKotlin(hiddenSince = "1.4")
public expect infix fun FloatArray.contentEquals(other: FloatArray): Boolean

/*
 * Returns `true` if the two specified arrays are
 * structurally* equal to one another.
 * i.e. contain the same number of the same elements in the same order.
 * The elements are compared for equality with the [equals][Any.equals] function.
 * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
 */
@SinceKotlin("1.1")
@DeprecatedSinceKotlin(hiddenSince = "1.4")
public expect infix fun DoubleArray.contentEquals(other: DoubleArray): Boolean

/*
 * Returns `true` if the two specified arrays are
 * structurally* equal to one another.
 * i.e. contain the same number of the same elements in the same order.
 * The elements are compared for equality with the [equals][Any.equals] function.
 * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
 */
@SinceKotlin("1.1")
@DeprecatedSinceKotlin(hiddenSince = "1.4")
public expect infix fun BooleanArray.contentEquals(other: BooleanArray): Boolean
```
means that 'NaN' is equal to itself and '-0.0' is not equal to '0.0'.

*\n@SinceKotlin("1.4")
public expect infix fun CharArray.contentEquals(other: CharArray): Boolean

*\n**\n* Returns `true` if the two specified arrays are *structurally* equal to one another, \n* i.e. contain the same number of the same elements in the same order.

*\n* \n* The elements are compared for equality with the [equals][Any.equals] function.

*\n For floating point numbers it means that 'NaN' is equal to itself and '-0.0' is not equal to '0.0'.

*\n@SinceKotlin("1.4")
public expect infix fun ByteArray.contentEquals(other: ByteArray?): Boolean

*\n**\n* Returns `true` if the two specified arrays are *structurally* equal to one another, \n* i.e. contain the same number of the same elements in the same order.

*\n* \n* The elements are compared for equality with the [equals][Any.equals] function.

*\n For floating point numbers it means that 'NaN' is equal to itself and '-0.0' is not equal to '0.0'.

*\n@SinceKotlin("1.4")
public expect infix fun ShortArray.contentEquals(other: ShortArray?): Boolean

*\n**\n* Returns `true` if the two specified arrays are *structurally* equal to one another, \n* i.e. contain the same number of the same elements in the same order.

*\n* \n* The elements are compared for equality with the [equals][Any.equals] function.

*\n For floating point numbers it means that 'NaN' is equal to itself and '-0.0' is not equal to '0.0'.

*\n@SinceKotlin("1.4")
public expect infix fun IntArray.contentEquals(other: IntArray?): Boolean

*\n**\n* Returns `true` if the two specified arrays are *structurally* equal to one another, \n* i.e. contain the same number of the same elements in the same order.

*\n* \n* The elements are compared for equality with the [equals][Any.equals] function.

*\n For floating point numbers it means that 'NaN' is equal to itself and '-0.0' is not equal to '0.0'.

*\n@SinceKotlin("1.4")
public expect infix fun LongArray.contentEquals(other: LongArray?): Boolean

*\n**\n* Returns `true` if the two specified arrays are *structurally* equal to one another, \n* i.e. contain the same number of the same elements in the same order.

*\n* \n* The elements are compared for equality with the [equals][Any.equals] function.

*\n For floating point numbers it means that 'NaN' is equal to itself and '-0.0' is not equal to '0.0'.

*\n@SinceKotlin("1.4")
public expect infix fun FloatArray.contentEquals(other: FloatArray?): Boolean

*\n**\n* Returns `true` if the two specified arrays are *structurally* equal to one another, \n* i.e. contain the same number of the same elements in the same order.

*\n* \n* The elements are compared for equality with the [equals][Any.equals] function.

*\n For floating point numbers it means that 'NaN' is equal to itself and '-0.0' is not equal to '0.0'.

*\n@SinceKotlin("1.4")
public expect infix fun DoubleArray.contentEquals(other: DoubleArray?): Boolean

*\n**\n* Returns `true` if the two specified arrays are *structurally* equal to one another, \n* i.e. contain the same number of the same elements in the same order.

*\n* \n* The elements are compared for equality with the [equals][Any.equals] function.

*\n For floating point numbers it means that 'NaN' is equal to itself and '-0.0' is not equal to '0.0'.

*\n@SinceKotlin("1.4")
public expect infix fun BooleanArray.contentEquals(other: BooleanArray?): Boolean

*\n**\n* Returns `true` if the two specified arrays are *structurally* equal to one another, \n* i.e. contain the same number of the same elements in the same order.

*\n* \n* The elements are compared for equality with the [equals][Any.equals] function.

*\n For floating point numbers it means that 'NaN' is equal to itself and '-0.0' is not equal to '0.0'.

*\n@SinceKotlin("1.4")
public expect infix fun CharArray.contentHashCode(): Int

*\n@SinceKotlin("1.1")
@DeprecatedSinceKotlin(hiddenSince = "1.4")
public expect fun CharArray.contentHashCode(): Int

*\n**\n* Returns a hash code based on the contents of this array as if it is [List].

*\n@SinceKotlin("1.1")
@DeprecatedSinceKotlin(hiddenSince = "1.4")
public expect fun ByteArray.contentHashCode(): Int

*\n**\n* Returns a hash code based on the contents of this array as if it is [List].

*\n@SinceKotlin("1.1")
@DeprecatedSinceKotlin(hiddenSince = "1.4")
public expect fun
Open Source Used In security_automation identity-guest-cs 1216

ShortArray.contentHashCode(): Int\n**\n * Returns a hash code based on the contents of this array as if it is [List].

IntArray.contentHashCode(): Int\n**\n * Returns a hash code based on the contents of this array as if it is [List].

LongArray.contentHashCode(): Int\n**\n * Returns a hash code based on the contents of this array as if it is [List].

FloatArray.contentHashCode(): Int\n**\n * Returns a hash code based on the contents of this array as if it is [List].

DoubleArray.contentHashCode(): Int\n**\n * Returns a hash code based on the contents of this array as if it is [List].

BooleanArray.contentHashCode(): Int\n**\n * Returns a hash code based on the contents of this array as if it is [List].

CharArray.contentHashCode(): Int\n**\n * Returns a hash code based on the contents of this array as if it is [List].

ByteArray.contentHashCode(): Int\n**\n * Returns a string representation of the contents of this array as if it is [List].

ShortArray.contentToString(): String\n**\n * Returns a string representation of the contents of the specified array as if it is [List].

IntArray.contentToString(): String\n**\n * Returns a string representation of the contents of the specified array as if it is [List].

LongArray.contentToString(): String\n**\n * Returns a string representation of the contents of the specified array as if it is [List].

FloatArray.contentToString(): String\n**\n * Returns a string representation of the contents of the specified array as if it is [List].

DoubleArray.contentToString(): String\n**\n * Returns a string representation of the contents of the specified array as if it is [List].

BooleanArray.contentToString(): String\n**\n * Returns a string representation of the contents of the specified array as if it is [List].

CharArray.contentToString(): String\n**\n * Returns a string representation of the contents of the specified array as if it is [List].

ByteArray.contentToString(): String\n**\n * Returns a string representation of the contents of the specified array as if it is [List].
if it is [List].

```java
@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.")
@SinceKotlin("1.1")
public expect fun LongArray.contentToString(): String

@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.")
@SinceKotlin("1.1")
public expect fun FloatArray.contentToString(): String

@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.")
@SinceKotlin("1.1")
public expect fun DoubleArray.contentToString(): String

@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.")
@SinceKotlin("1.1")
public expect fun BooleanArray.contentToString(): String

@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.")
@SinceKotlin("1.1")
public expect fun CharArray.contentToString(): String

@SinceKotlin("1.4")
public expect fun <T> Array<out T>?.contentToString(): String

@SinceKotlin("1.4")
public expect fun ByteArray?.contentToString(): String

@SinceKotlin("1.4")
public expect fun ShortArray?.contentToString(): String

@SinceKotlin("1.4")
public expect fun IntArray?.contentToString(): String

@SinceKotlin("1.4")
public expect fun LongArray?.contentToString(): String

@SinceKotlin("1.4")
public expect fun FloatArray?.contentToString(): String

@SinceKotlin("1.4")
public expect fun DoubleArray?.contentToString(): String

@SinceKotlin("1.4")
public expect fun BooleanArray?.contentToString(): String

@SinceKotlin("1.4")
public expect fun CharArray?.contentToString(): String
```

Copies this array or its subrange into the [destination] array and returns that array.

```
@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.")
@SinceKotlin("1.1")
public expect fun DoubleArray.contentToString(): String
```
array indices or when \`startIndex > endIndex\`.

* @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset],

* or when that index is out of the [destination] array indices range.

* @return return the [destination] array.

* __@SinceKotlin(\"1.3\")__

*public expect fun <T> Array<out T>.copyInto(destination: Array<T>, destinationOffset: Int = 0, startIndex: Int = Int = size): Array<T>

* @param destination the array to copy to.

* @param destinationOffset the position in the [destination] array to copy to, 0 by default.

* @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.

* @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.

* @return the [destination] array.

* __@SinceKotlin(\"1.3\")__

*public expect fun ByteArray.copyInto(destination: ByteArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): ByteArray

* @return the [destination] array.

* __@SinceKotlin(\"1.3\")__

*public expect fun ShortArray.copyInto(destination: ShortArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): ShortArray

* @return the [destination] array.

* __@SinceKotlin(\"1.3\")__

*public expect fun IntArray.copyInto(destination: IntArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): IntArray

* @return the [destination] array.

* __@SinceKotlin(\"1.3\")__

*public expect fun LongArray.copyInto(destination: LongArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): LongArray

* @return the [destination] array.

* __@SinceKotlin(\"1.3\")__

*public expect fun IntArray.copyInto(destination: IntArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): IntArray

* @return the [destination] array.

* __@SinceKotlin(\"1.3\")__

*public expect fun ShortArray.copyInto(destination: ShortArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): ShortArray

* @return the [destination] array.

* __@SinceKotlin(\"1.3\")__

*public expect fun IntArray.copyInto(destination: IntArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): IntArray

* @return the [destination] array.

* __@SinceKotlin(\"1.3\")__

*public expect fun LongArray.copyInto(destination: LongArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): LongArray

* @return the [destination] array.

* __@SinceKotlin(\"1.3\")__

*public expect fun IntArray.copyInto(destination: IntArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): IntArray

* @return the [destination] array.

* __@SinceKotlin(\"1.3\")__

*public expect fun ShortArray.copyInto(destination: ShortArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): ShortArray

* @return the [destination] array.

* __@SinceKotlin(\"1.3\")__

*public expect fun IntArray.copyInto(destination: IntArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): IntArray

* @return the [destination] array.

* __@SinceKotlin(\"1.3\")__

*public expect fun LongArray.copyInto(destination: LongArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): LongArray

* @return the [destination] array.

* __@SinceKotlin(\"1.3\")__

*public expect fun IntArray.copyInto(destination: IntArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): IntArray

* @return the [destination] array.
destination range.

* @param destination the array to copy to.
* @param destinationOffset the position in the [destination] array to copy to, 0 by default.
* @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.
* @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.
* @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `{startIndex > endIndex}`.
* @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset], or when that index is out of the [destination] array indices range.

* @SinceKotlin("1.3")
* @return the [destination] array.

@SinceKotlin("1.3")
public expect fun FloatArray.copyInto(destination: FloatArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): FloatArray

@SinceKotlin("1.3")
public expect fun DoubleArray.copyInto(destination: DoubleArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): DoubleArray

@SinceKotlin("1.3")
public expect fun BooleanArray.copyInto(destination: BooleanArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): BooleanArray

@SinceKotlin("1.3")
public expect fun CharArray.copyInto(destination: CharArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): CharArray

@SinceKotlin("1.3")
@Suppress("NO_ACTUAL_FOR_EXPECT")
public expect fun <T> Array<T>.copyOf(): Array<T>

@SinceKotlin("1.3")
public expect fun ByteArray.copyOf(): ByteArray

@SinceKotlin("1.3")
public expect fun ShortArray.copyOf(): ShortArray
samples.collections.Arrays.CopyOfOperations.copyOf(n) *
*\n\n\n@sample
samples.collections.Arrays.CopyOfOperations.copyOf(n) *
*\n\nLongArray\n\n**\n* Returns new array which is a copy of the original array.
*\n\n@sample
samples.collections.Arrays.CopyOfOperations.copyOf(n) *
*\n\nFloatArray\n\n**\n* Returns new array which is a copy of the original array.
*\n\n@sample
samples.collections.Arrays.CopyOfOperations.copyWith(n) *
*\n\nDoubleArray\n\n**\n* Returns new array which is a copy of the original array.
*\n\n@sample
samples.collections.Arrays.CopyOfOperations.copyWith(n) *
*\n\nBooleanArray\n\n**\n* Returns new array which is a copy of the original array.
*\n\n@sample
samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf(n) *
*\n\nByte\n\n\n**\n* Returns new array which is a copy of the original array,
resized to the given [newSize].
*\n\n@sample
samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf(n) *
*\n\nShortArray\n\n**\n* Returns new array which is a copy of the original array,
resized to the given [newSize].
*\n\n@sample
samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf(n) *
*\n\nIntArray\n\n**\n* Returns new array which is a copy of the original array,
resized to the given [newSize].
*\n\n@sample
samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf(n) *
*\n\nLongArray\n\n**\n* Returns new array which is a copy of the original array,
resized to the given [newSize].
*\n\n@sample
samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf(n) *
*\n\nIntArray\n\n**\n* Returns new array which is a copy of the original array,
resized to the given [newSize].
*\n\n@sample
samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf(n) *
*\n\nCharArray\n\n**\n* Returns new array which is a copy of the original array,
resized to the given [newSize].
*\n\n@sample
samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf(n) *
*\n\nByteArray\n\n**\n* Returns new array which is a copy of the original array,
resized to the given [newSize].
*\n\n@sample
samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf(n) *
*\n\nShortArray\n\n**\n* Returns new array which is a copy of the original array,
resized to the given [newSize].
*\n\n@sample
samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf(n) *
*\n\nIntArray\n\n**\n* Returns new array which is a copy of the original array,
resized to the given [newSize].
*\n\n@sample
samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf(n) *
*\n\nLongArray\n\n**\n* Returns new array which is a copy of the original array,
resized to the given [newSize].
*\n\n@sample
samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf(n) *
*\n\nIntArray\n\n**\n* Returns new array which is a copy of the original array,
resized to the given [newSize].
*\n\n@sample
samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf(n) *
*\n\nShortArray\n\n**\n* Returns new array which is a copy of the original array,
resized to the given [newSize].
*\n\n@sample
samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf(n) *
*\n\nIntArray\n\n**\n* Returns new array which is a copy of the original array,
resized to the given [newSize].
*\n\n@sample
samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf(n) *
*\n\nShortArray\n\n**\n* Returns new array which is a copy of the original array,
resized to the given [newSize].
*\n\n@sample
samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf(n) *
*\n\nIntArray\n\n**\n* Returns new array which is a copy of the original array,
resized to the given [newSize].
*\n\n@sample
samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf(n) *
*\n\nShortArray\n\n**\n* Returns new array which is a copy of the original array,
resized to the given [newSize].
*\n\n@sample
samplescollections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf(n) *
*\n\nIntArray\n\n**\n* Returns new array which is a copy of the original array,
resized to the given [newSize].
*\n\n@sample
samplescollections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf(n) *
*\n\nShortArray\n\n**\n* Returns new array which is a copy of the original array,
resized to the given [newSize].
*\n\n@sample
samplescollections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf(n) *
*\n\nIntArray\n\n**\n* Returns new array which is a copy of the original array,
resized to the given [newSize].
*\n\n@sample
samplescollections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf(n) *
*\n\nShortArray\n\n**\n* Returns new array which is a copy of the original array,
resized to the given [newSize].
*\n\n@sample
samplescollections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf(n) *
*\n\nIntArray\n\n**\n* Returns new array which is a copy of the original array,
resized to the given [newSize].
*\n\n@sample
samplescollections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf(n) *
*\n\nShortArray\n\n**\n* Returns new array which is a copy of the original array,
resized to the given [newSize].
*\n\n@sample
samplescollections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf(n) *
*\n\nIntArray\n\n**\n* Returns new array which is a copy of the original array,
resized to the given [newSize].
*\n\n@sample
samplescollections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf(n) *
*\n\nShortArray\n\n**\n* Returns new array which is a copy of the original array,
resized to the given [newSize].
*\n\n@sample
samplescollections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf(n) *
*\n\nIntArray\n\n**\n* Returns new array which is a copy of the original array,
array, resized to the given [newSize].
 * The copy is either truncated or padded at the end with null char (\u0000) values if necessary.
 * If [newSize] is less than the size of the original array, the copy array is truncated to the
 [newSize].
 * If [newSize] is greater than the size of the original array, the extra elements in the copy array are
 filled with null char (\u0000) values.
 * @sample samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf
 */
public expect fun CharArray.copyOf(newSize: Int): CharArray

/**
 * Returns new array which is a copy of the original array, resized to the given [newSize].
 * The copy is either truncated or padded at the end with 'null' values if necessary.
 * If [newSize] is less than the size of the original array, the copy array is truncated to the
 [newSize].
 * If [newSize] is greater than the size of the original array, the extra elements in the copy array are
 filled with 'null' values.
 * @sample samples.collections.Arrays.CopyOfOperations.resizingCopyOf
 */
@Suppress("NO_ACTUAL_FOR_EXPECT")
public expect fun <T> Array<T>.copyOf(newSize: Int): Array<T>

/**
 * Returns a new array which is a copy of the specified range of the original array.
 * The copy is either truncated or padded at the end of the range (exclusive) to copy.
 * @param fromIndex the start of the range (inclusive) to copy.
 * @param toIndex the end of the range (exclusive) to copy.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
@Suppress("NO_ACTUAL_FOR_EXPECT")
public expect fun ByteArray.copyOfRange(fromIndex: Int, toIndex: Int): ByteArray

/**
 * Returns a new array which is a copy of the specified range of the original array.
 * @param fromIndex the start of the range (inclusive) to copy.
 * @param toIndex the end of the range (exclusive) to copy.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
public expect fun ShortArray.copyOfRange(fromIndex: Int, toIndex: Int): ShortArray

/**
 * Returns a new array which is a copy of the specified range of the original array.
 * @param fromIndex the start of the range (inclusive) to copy.
 * @param toIndex the end of the range (exclusive) to copy.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
public expect fun IntArray.copyOfRange(fromIndex: Int, toIndex: Int): IntArray

/**
 * Returns a new array which is a copy of the specified range of the original array.
 * @param fromIndex the start of the range (inclusive) to copy.
 * @param toIndex the end of the range (exclusive) to copy.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
public expect fun LongArray.copyOfRange(fromIndex: Int, toIndex: Int): LongArray

/**
 * Returns a new array which is a copy of the specified range of the original array.
 * @param fromIndex the start of the range (inclusive) to copy.
 * @param toIndex the end of the range (exclusive) to copy.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
public expect fun FloatArray.copyOfRange(fromIndex: Int, toIndex: Int): FloatArray

/**
 * Returns a new array which is a copy of the specified range of the original array.
 * @param fromIndex the start of the range (inclusive) to copy.
 * @param toIndex the end of the range (exclusive) to copy.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
public expect fun DoubleArray.copyOfRange(fromIndex: Int, toIndex: Int): DoubleArray

if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.

* @throws
IllegalArgumentException if [fromIndex] is greater than [toIndex].
* @throws
IllegalStateException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
* @throws
IllegalStateException if [fromIndex] is greater than [toIndex].
BooleanArray.fill(element: Boolean, fromIndex: Int = 0, toIndex: Int = size): Unit

*An@SinceKotlin("1.3")@public fun CharArray.fill(element: Char, fromIndex: Int = 0, toIndex: Int = size): Unit

/**
 * Fills this array or its subrange with the specified [element] value.
 * @param fromIndex the start of the range (inclusive) to fill, 0 by default.
 * @param toIndex the end of the range (exclusive) to fill, size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 */

@SinceKotlin("1.3")
@public fun CharArray.fill(element: Char, fromIndex: Int = 0, toIndex: Int = size): Unit

/**
 * Returns the range of valid indices for the array.
 */
@public fun BooleanArray.indices: IntRange
    get() = IntRange(0, lastIndex)

/**
 * Returns the range of valid indices for the array.
 */
@public fun CharArray.indices: IntRange
    get() = IntRange(0, lastIndex)

/**
 * Returns the range of valid indices for the array.
 */
@public fun ByteArray.indices: IntRange
    get() = IntRange(0, lastIndex)

/**
 * Returns the range of valid indices for the array.
 */
@public fun ByteArray.indices: IntRange
    get() = IntRange(0, lastIndex)

/**
 * Returns the range of valid indices for the array.
 */
@public fun ShortArray.indices: IntRange
    get() = IntRange(0, lastIndex)

/**
 * Returns the range of valid indices for the array.
 */
@public fun IntArray.indices: IntRange
    get() = IntRange(0, lastIndex)

/**
 * Returns the range of valid indices for the array.
 */
@public fun LongArray.indices: IntRange
    get() = IntRange(0, lastIndex)

/**
 * Returns the range of valid indices for the array.
 */
@public fun FloatArray.indices: IntRange
    get() = IntRange(0, lastIndex)

/**
 * Returns the range of valid indices for the array.
 */
@public fun DoubleArray.indices: IntRange
    get() = IntRange(0, lastIndex)

/**
 * Returns the range of valid indices for the array.
 */
@public fun BooleanArray.fill(element: Boolean, fromIndex: Int = 0, toIndex: Int = size): Unit

*/

@SinceKotlin("1.3")
@public fun ByteArray.fill(element: Byte, fromIndex: Int = 0, toIndex: Int = size): Unit

/**
 * Returns the range of valid indices for the array.
 */
@public fun ByteArray.indices: IntRange
    get() = IntRange(0, lastIndex)

/**
 * Returns the range of valid indices for the array.
 */
@public fun ByteArray.indices: IntRange
    get() = IntRange(0, lastIndex)

/**
 * Returns the range of valid indices for the array.
 */
@public fun ShortArray.indices: IntRange
    get() = IntRange(0, lastIndex)

/**
 * Returns the range of valid indices for the array.
 */
@public fun IntArray.indices: IntRange
    get() = IntRange(0, lastIndex)

/**
 * Returns the range of valid indices for the array.
 */
array.

```kotlin
*//public val IntArray.lastIndex: Int
    get() = size - 1
*//public val LongArray.lastIndex: Int
    get() = size - 1
*//public val FloatArray.lastIndex: Int
    get() = size - 1
*//public val DoubleArray.lastIndex: Int
    get() = size - 1
*//public val BooleanArray.lastIndex: Int
    get() = size - 1
*//public val CharArray.lastIndex: Int
    get() = size - 1
```

```kotlin
/**
 * Returns the last valid index for the
 * array.
 */
public val IntArray.lastIndex: Int
    get() = size - 1

/**
 * Returns the last valid index for the
 * array.
 */
public val LongArray.lastIndex: Int
    get() = size - 1

/**
 * Returns the last valid index for the
 * array.
 */
public val FloatArray.lastIndex: Int
    get() = size - 1

/**
 * Returns the last valid index for the
 * array.
 */
public val DoubleArray.lastIndex: Int
    get() = size - 1

/**
 * Returns the last valid index for the
 * array.
 */
public val BooleanArray.lastIndex: Int
    get() = size - 1

/**
 * Returns the last valid index for the
 * array.
 */
public val CharArray.lastIndex: Int
    get() = size - 1
```

```kotlin
/**
 * Returns an array containing all elements of the original array and then the given [element].
 */
@Suppress("NO_ACTUAL_FOR_EXPECT")
public expect operator fun <T> Array<T>.plus(element: T): Array<T>

/**
 * Returns an array containing all elements of the original array and then the given [element].
 */
public expect operator fun ByteArray.plus(element: Byte): ByteArray

/**
 * Returns an array containing all elements of the original array and then the given [element].
 */
public expect operator fun ShortArray.plus(element: Short): ShortArray

/**
 * Returns an array containing all elements of the original array and then the given [element].
 */
public expect operator fun IntArray.plus(element: Int): IntArray

/**
 * Returns an array containing all elements of the original array and then the given [element].
 */
public expect operator fun LongArray.plus(element: Long): LongArray

/**
 * Returns an array containing all elements of the original array and then the given [element].
 */
public expect operator fun FloatArray.plus(element: Float): FloatArray

/**
 * Returns an array containing all elements of the original array and then the given [element].
 */
public expect operator fun DoubleArray.plus(element: Double): DoubleArray

/**
 * Returns an array containing all elements of the original array and then the given [element].
 */
public expect operator fun BooleanArray.plus(element: Boolean): BooleanArray

/**
 * Returns an array containing all elements of the original array and then the given [element].
 */
public expect operator fun CharArray.plus(element: Char): CharArray

/**
 * Returns an array containing all elements of the original array and then all elements of the
 * given [elements].
 */
@Suppress("NO_ACTUAL_FOR_EXPECT")
public expect operator fun <T> Array<T>.plus(elements: Collection<T>): Array<T>

/**
 * Returns an array containing all elements of the original array and then all elements of the
 * given [elements].
 */
public expect operator fun ByteArray.plus(elements: Collection<Byte>): ByteArray

/**
 * Returns an array containing all elements of the original array and then all elements of the
 * given [elements].
 */
public expect operator fun ShortArray.plus(elements: Collection<Short>): ShortArray

/**
 * Returns an array containing all elements of the original array and then all elements of the
 * given [elements].
 */
public expect operator fun IntArray.plus(elements: Collection<Int>): IntArray

/**
 * Returns an array containing all elements of the original array and then all elements of the
 * given [elements].
 */
public expect operator fun LongArray.plus(elements: Collection<Long>): LongArray

/**
 * Returns an array containing all elements of the original array and then all elements of the
 * given [elements].
 */
public expect operator fun FloatArray.plus(elements: Collection<Float>): FloatArray

/**
 * Returns an array containing all elements of the original array and then all elements of the
 * given [elements].
 */
public expect operator fun DoubleArray.plus(elements: Collection<Double>): DoubleArray

/**
 * Returns an array containing all elements of the original array and then all elements of the
 * given [elements].
 */
public expect operator fun BooleanArray.plus(elements: Collection<Boolean>): BooleanArray

/**
 * Returns an array containing all elements of the original array and then all elements of the
 * given [elements].
 */
public expect operator func Array<T>.plus(elements: Array<T>): Array<T>

/**
 * Returns an array containing all elements of the original array and then all elements of the
 * given [elements].
 */
public expect operator fun ByteArray.plus(elements: ByteArray): ByteArray

/**
 * Returns an array containing all elements of the original array and then all elements of the
 * given [elements].
 */
public expect operator func ShortArray.plus(elements: ShortArray): ShortArray
```

Open Source Used In security_automation identity-guest-cs 1224
Returns an array containing all elements of the original array and then all elements of the given [elements] array.

public expect operator fun LongArray.plus(elements: LongArray): LongArray

Returns an array containing all elements of the original array and then all elements of the given [elements] array.

public expect operator fun FloatArray.plus(elements: FloatArray): FloatArray

Returns an array containing all elements of the original array and then all elements of the given [elements] array.

public expect operator fun DoubleArray.plus(elements: DoubleArray): DoubleArray

Returns an array containing all elements of the original array and then all elements of the given [elements] array.

public expect operator fun BooleanArray.plus(elements: BooleanArray): BooleanArray
IntArray.sort(fromIndex: Int = 0, toIndex: Int = size): Unit

/**
 * Sorts a range in the array in-place.
 * @param fromIndex the start of the range (inclusive) to sort, 0 by default.
 * @param toIndex the end of the range (exclusive) to sort, size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
@SinceKotlin("1.4")
public expect fun IntArray.sort(fromIndex: Int = 0, toIndex: Int = size): Unit

FloatArray.sort(fromIndex: Int = 0, toIndex: Int = size): Unit

/**
 * Sorts a range in the array in-place.
 * @param fromIndex the start of the range (inclusive) to sort, 0 by default.
 * @param toIndex the end of the range (exclusive) to sort, size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
@SinceKotlin("1.4")
public expect fun FloatArray.sort(fromIndex: Int = 0, toIndex: Int = size): Unit

DoubleArray.sort(fromIndex: Int = 0, toIndex: Int = size): Unit

/**
 * Sorts a range in the array in-place.
 * @param fromIndex the start of the range (inclusive) to sort, 0 by default.
 * @param toIndex the end of the range (exclusive) to sort, size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
@SinceKotlin("1.4")
public expect fun DoubleArray.sort(fromIndex: Int = 0, toIndex: Int = size): Unit

CharArray.sort(fromIndex: Int = 0, toIndex: Int = size): Unit

/**
 * Sorts elements of the array in-place.
 * The elements are sorted descending according to their natural sort order.
 * @param fromIndex the start of the range (inclusive) to sort.
 * @param toIndex the end of the range (exclusive) to sort.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
@SinceKotlin("1.4")
public fun <T : Comparable<T>> Array<out T>.sortDescending(fromIndex: Int,
toIndex: Int): Unit

ByteArray.sort(fromIndex: Int = 0, toIndex: Int = size): Unit

/**
 * Sorts elements of the array in-place.
 * The elements are sorted descending according to their natural sort order.
 * @param fromIndex the start of the range (inclusive) to sort.
 * @param toIndex the end of the range (exclusive) to sort.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
@SinceKotlin("1.4")
public fun ByteArray.sortDescending(fromIndex: Int, toIndex: Int): Unit

ShortArray.sort(fromIndex: Int = 0, toIndex: Int = size): Unit

/**
 * Sorts elements of the array in-place.
 * The elements are sorted descending according to their natural sort order.
 * @param fromIndex the start of the range (inclusive) to sort.
 * @param toIndex the end of the range (exclusive) to sort.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
@SinceKotlin("1.4")
public fun ShortArray.sortDescending(fromIndex: Int, toIndex: Int): Unit

IntArray.sortDescending(fromIndex: Int = 0, toIndex: Int = size): Unit

/**
 * Sorts elements of the array in-place.
 * The elements are sorted descending according to their natural sort order.
 * @param fromIndex the start of the range (inclusive) to sort.
 * @param toIndex the end of the range (exclusive) to sort.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
@SinceKotlin("1.4")
public fun IntArray.sortDescending(fromIndex: Int, toIndex: Int): Unit
range in-place.

The elements are sorted descending according to their natural sort order.

@throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.

@param fromIndex the start of the range (inclusive) to sort.

@throws IllegalArgumentExcept...
of this primitive array.
*/

//fixme: This implementation is the same as in the standard library.

/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */

fun ByteArray.toTypedArray(): Array<Byte> = throw UnsupportedOperationException()

/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */

fun ShortArray.toTypedArray(): Array<Short> = throw UnsupportedOperationException()

/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */

fun IntArray.toTypedArray(): Array<Int> = throw UnsupportedOperationException()

/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */

fun LongArray.toTypedArray(): Array<Long> = throw UnsupportedOperationException()

/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */

fun FloatArray.toTypedArray(): Array<Float> = throw UnsupportedOperationException()

/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */

fun DoubleArray.toTypedArray(): Array<Double> = throw UnsupportedOperationException()

/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */

fun BooleanArray.toTypedArray(): Array<Boolean> = throw UnsupportedOperationException()

/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */

fun CharArray.toTypedArray(): Array<Char> = throw UnsupportedOperationException()

/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */

fun <T, K, V> Array<out T>.associate(transform: (T) -> Pair<K, V>): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateTo(LinkedHashMap<K, V>(capacity), transform) as Map<K, V>*
}

/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */

fun ByteArray.associate(transform: (Byte) -> Pair<K, V>): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateTo(LinkedHashMap<K, V>(capacity), transform) as Map<K, V>*
}

/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */

fun ShortArray.associate(transform: (Short) -> Pair<K, V>): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateTo(LinkedHashMap<K, V>(capacity), transform) as Map<K, V>*
}

/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */

fun IntArray.associate(transform: (Int) -> Pair<K, V>): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateTo(LinkedHashMap<K, V>(capacity), transform) as Map<K, V>*
}

/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */

fun LongArray.associate(transform: (Long) -> Pair<K, V>): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateTo(LinkedHashMap<K, V>(capacity), transform) as Map<K, V>*
}

/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */

fun FloatArray.associate(transform: (Float) -> Pair<K, V>): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateTo(LinkedHashMap<K, V>(capacity), transform) as Map<K, V>*
}

*/

elements of the given array. If any of two pairs would have the same key the last one gets added to the map. The returned map preserves the entry iteration order of the original array. If any of two pairs would have the same key the last one gets added to the map. The returned map preserves the entry iteration order of the original array. If any of two pairs would have the same key the last one gets added to the map. The returned map preserves the entry iteration order of the original array. The returned map preserves the entry iteration order of the original array. The returned map preserves the entry iteration order of the original array.
order of the original array.

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesBy

LongArray.associateBy(keySelector: (Long) -> K): Map<K, Long> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, Long>(capacity),
        keySelector).@sample
}

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesBy

FloatArray.associateBy(keySelector: (Float) -> K): Map<K, Float> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, Float>(capacity),
        keySelector).@sample
}

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesBy

DoubleArray.associateBy(keySelector: (Double) -> K): Map<K, Double> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, Double>(capacity),
        keySelector).@sample
}

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesBy

CharArray.associateBy(keySelector: (Char) -> K): Map<K, Char> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, Char>(capacity),
        keySelector).@sample
}

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesBy

ByteArray.associateBy(keySelector: (Byte) -> K, valueTransform: (Byte) -> V): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, V>(capacity),
        keySelector, valueTransform).@sample
}

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesBy

TArray.associateBy(keySelector: (T) -> K, valueTransform: (T) -> V): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, V>(capacity),
        keySelector, valueTransform).@sample
}

Open Source Used In security_automation identity-guest-cs 1230
have the same key returned by [keySelector] the last one gets added to the map.

The returned map preserves the entry iteration order of the original array.

```kotlin
fun <K, V> ShortArray.associateBy(keySelector: (Short) -> K, valueTransform: (Short) -> V): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, V>(capacity), keySelector, valueTransform)
}
```

```kotlin
fun <K, V> IntArray.associateBy(keySelector: (Int) -> K, valueTransform: (Int) -> V): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, V>(capacity), keySelector, valueTransform)
}
```

```kotlin
fun <K, V> LongArray.associateBy(keySelector: (Long) -> K, valueTransform: (Long) -> V): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, V>(capacity), keySelector, valueTransform)
}
```

```kotlin
fun <K, V> FloatArray.associateBy(keySelector: (Float) -> K, valueTransform: (Float) -> V): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, V>(capacity), keySelector, valueTransform)
}
```

```kotlin
fun <K, V> DoubleArray.associateBy(keySelector: (Double) -> K, valueTransform: (Double) -> V): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, V>(capacity), keySelector, valueTransform)
}
```

```kotlin
fun <K, V> BooleanArray.associateBy(keySelector: (Boolean) -> K, valueTransform: (Boolean) -> V): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, V>(capacity), keySelector, valueTransform)
}
```

```kotlin
fun <K, V> CharArray.associateBy(keySelector: (Char) -> K, valueTransform: (Char) -> V): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, V>(capacity), keySelector, valueTransform)
}
```

```kotlin
fun <K, V> Array<*>.associateBy(keySelector: (Array<*>) -> K, valueTransform: (Array<*>) -> V): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, V>(capacity), keySelector, valueTransform)
}
```
pairs, where key is provided by the [keySelector] function applied to each element of the given array and value is the element itself. If any two elements would have the same key returned by [keySelector] the last one gets added to the map.

```kotlin
@sample
samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByTo
```
return destination

* Populates and returns the [destination] mutable map with key-value pairs,
* where key is provided by the [keySelector] function applied to each element of the given array
* and value is the element itself.
* If any two elements would have the same key returned by [keySelector] the last one gets added to the map.

* Populates and returns the [destination] mutable map with key-value pairs,
* where key is provided by the [keySelector] function and
* and value is provided by the [valueTransform] function applied to elements of the given array.
* If any two elements would have the same key returned by [keySelector] the last one gets added to the map.

* * *
value Transform(element)\n | n return destination:n\n/*n * Populates and returns the [destination] mutable map with key-value pairs,n * where key is provided by the [keySelector] function andn * and value is provided by the [value Transform] function applied to elements of the given array.n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.n * \n * @sample

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByToWithValueTransform
*

/**
 * Populates and returns the [destination] mutable map with key-value pairs,n * where key is provided by the [keySelector] function andn * and value is provided by the [value Transform] function applied to elements of the given array.n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.n * \n * @sample

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByToWithValueTransform
*

*/

//n 

public inline fun <K, V, M : MutableMap<n K, in V>> DoubleArray.associateByTo(destination: M, keySelector:

(Double) -> K, value Transform: (Double) -> V): M {n \n for (element in this) \n
destination.put(keySelector(element), value Transform(element))\n | n return destination:n\n/*n * Populates and returns the [destination] mutable map with key-value pairs,n * where key is provided by the [keySelector] function andn * and value is provided by the [value Transform] function applied to elements of the given array.n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.n * \n * @sample

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByToWithValueTransform
*

/**
 * Populates and returns the [destination] mutable map with key-value pairs,n * where key is provided by the [keySelector] function andn * and value is provided by the [value Transform] function applied to elements of the given array.n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.n * \n * @sample

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByToWithValueTransform
*

*/

//n 

public inline fun <K, V, M : MutableMap<n K, in V>> BooleanArray.associateByTo(destination: M,

keySelector: (Boolean) -> K, value Transform: (Boolean) -> V): M {n \n for (element in this) \n
destination.put(keySelector(element), value Transform(element))\n | n return destination:n\n/*n * Populates and returns the [destination] mutable map with key-value pairs,n * where key is provided by the [keySelector] function andn * and value is provided by the [value Transform] function applied to elements of the given array.n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.n * \n * @sample

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByToWithValueTransform
*

/**
 * Populates and returns the [destination] mutable map with key-value pairs,n * where key is provided by the [keySelector] function andn * and value is provided by the [value Transform] function applied to elements of the given array.n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.n * \n * @sample

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByToWithValueTransform
*/

/**
 * Populates and returns the [destination] mutable map with key-value pairs
 * provided by [transform] function applied to each
 * element of the given array.n * \n * If any of two pairs would have the same key the last one gets added to the map.n * \n * @sample

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesTo
*

/**
 * Populates and returns the [destination] mutable map with key-value pairs
 * provided by [transform] function applied to each
 * element of the given array.n * \n * If any of two pairs would have the same key the last one gets added to the map.n * \n * @sample

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesTo
*/

/**
 * Populates and returns the [destination] mutable map with key-value pairs
 * provided by [transform] function applied to each
 * element of the given array.n * \n * If any of two pairs would have the same key the last one gets added to the map.n * \n * @sample

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesTo
*/

/**
 * Populates and returns the [destination] mutable map with key-value pairs
 * provided by [transform] function applied to each
 * element of the given array.n * \n * If any of two pairs would have the same key the last one gets added to the map.n * \n * @sample

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesTo
*/

/**
 * Populates and returns the [destination] mutable map with key-value pairs
 * provided by [transform] function applied to each
 * element of the given array.n * \n * If any of two pairs would have the same key the last one gets added to the map.n * \n * @sample

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesTo
*/

/**
 * Populates and returns the [destination] mutable map with key-value pairs
 * provided by [transform] function applied to each
 * element of the given array.n * \n * If any of two pairs would have the same key the last one gets added to the map.n * \n * @sample

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesTo
*/

/**
 * Populates and returns the [destination] mutable map with key-value pairs
 * provided by [transform] function applied to each
 * element of the given array.n * \n * If any of two pairs would have the same key the last one gets added to the map.n * \n * @sample

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesTo
*/

/**
 * Populates and returns the [destination] mutable map with key-value pairs
 * provided by [transform] function applied to each
 * element of the given array.n * \n * If any of two pairs would have the same key the last one gets added to the map.n * \n * @sample

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesTo
*/

/**
 * Populates and returns the [destination] mutable map with key-value pairs
 * provided by [transform] function applied to each
 * element of the given array.n * \n * If any of two pairs would have the same key the last one gets added to the map.n * \n * @sample

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesTo
*/
Pair<K, V>): M {\n    for (element in this) {\n        destination += transform(element)\n    }\n    return destination\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs\n * provided by [transform] function applied to each element of the given array.\n * If any of two pairs would have the same key the last one gets added to the\n * map.\n */\n@sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesTo\n\nMutableMap<K, V>: Map<K, V>\n\nPair<K, V>): M {\n    for (element in this) {\n        destination += transform(element)\n    }\n    return destination\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs\n * provided by [transform] function applied to each element of the given array.\n * If any of two pairs would have the same key the last one gets added to the\n * map.\n */\n@sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesTo\n\nFloatArray: FloatArray\n\nassociateTo(destination: M, transform: (Float) \r
-> Pair<K, V>): M {\n    for (element in this) {\n        destination += transform(element)\n    }\n    return destination\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs\n * provided by [transform] function applied to each element of the given array.\n * If any of two pairs would have the same key the last one gets added to the\n * map.\n */\n@sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesTo\n\nDoubleArray: DoubleArray\n\nassociateTo(destination: M, transform: (Double) \r
-> Pair<K, V>): M {\n    for (element in this) {\n        destination += transform(element)\n    }\n    return destination\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs\n * provided by [transform] function applied to each element of the given array.\n * If any of two pairs would have the same key the last one gets added to the\n * map.\n */\n@sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesTo
IntArray.associateWith(valueSelector: (Int) -> V): Map<Int, V> { val result = LinkedHashmap<Int, V>(mapCapacity(size).coerceAtLeast(16)) return associateWithTo(result, valueSelector) }

*Licensed@SinceKotlin("1.4")@kotlin.internal.InlineOnly*unpublic inline fun <V> LongArray.associateWith(valueSelector: (Long) -> V): Map<Long, V> { val result = LinkedHashmap<Long, V>(mapCapacity(size).coerceAtLeast(16)) return associateWithTo(result, valueSelector) }

*Licensed@SinceKotlin("1.4")@kotlin.internal.InlineOnly*unpublic inline fun <V> FloatArray.associateWith(valueSelector: (Float) -> V): Map<Float, V> { val result = LinkedHashmap<Float, V>(mapCapacity(size).coerceAtLeast(16)) return associateWithTo(result, valueSelector) }

*Licensed@SinceKotlin("1.4")@kotlin.internal.InlineOnly*unpublic inline fun <V> DoubleArray.associateWith(valueSelector: (Double) -> V): Map<Double, V> { val result = LinkedHashmap<Double, V>(mapCapacity(size).coerceAtLeast(16)) return associateWithTo(result, valueSelector) }

*Licensed@SinceKotlin("1.4")@kotlin.internal.InlineOnly*unpublic inline fun <V> BooleanArray.associateWith(valueSelector: (Boolean) -> V): Map<Boolean, V> { val result = LinkedHashmap<Boolean, V>(mapCapacity(size).coerceAtMost(128).coerceAtLeast(16)) return associateWithTo(result, valueSelector) }

*Licensed@SinceKotlin("1.4")@kotlin.internal.InlineOnly*unpublic inline fun <V> CharArray.associateWith(valueSelector: (Char) -> V): Map<Char, V> { val result = LinkedHashmap<Char, V>(mapCapacity(size.coerceAtMost(128)).coerceAtLeast(16)) return associateWithTo(result, valueSelector) }

Open Source Used In security_automation identity-guest-cs 1236
ByteArray.associateWithTo(destination: M, valueSelector: (Byte) -> V): M {
    for (element in this) {
        destination.put(element, valueSelector(element))
    }
    return destination
}

/**
 * Populates and returns the [destination] mutable map with key-value pairs for each element of the given array, where key is the element itself and value is provided by the [valueSelector] function applied to that key.
 * If any two elements are equal, the last one overwrites the former value in the map.
 * @sample samples.collections.Collections.Transformations.associateWithTo
 */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun <V, M : MutableMap<in Byte, in V>> ShortArray.associateWithTo(destination: M, valueSelector: (Short) -> V): M {
    for (element in this) {
        destination.put(element, valueSelector(element))
    }
    return destination
}

/**
 * Populates and returns the [destination] mutable map with key-value pairs for each element of the given array, where key is the element itself and value is provided by the [valueSelector] function applied to that key.
 * If any two elements are equal, the last one overwrites the former value in the map.
 * @sample samples.collections.Collections.Transformations.associateWithTo
 */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun <V, M : MutableMap<in Short, in V>> IntArray.associateWithTo(destination: M, valueSelector: (Int) -> V): M {
    for (element in this) {
        destination.put(element, valueSelector(element))
    }
    return destination
}

/**
 * Populates and returns the [destination] mutable map with key-value pairs for each element of the given array, where key is the element itself and value is provided by the [valueSelector] function applied to that key.
 * If any two elements are equal, the last one overwrites the former value in the map.
 * @sample samples.collections.Collections.Transformations.associateWithTo
 */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun <V, M : MutableMap<in Short, in V>> LongArray.associateWithTo(destination: M, valueSelector: (Long) -> V): M {
    for (element in this) {
        destination.put(element, valueSelector(element))
    }
    return destination
}

/**
 * Populates and returns the [destination] mutable map with key-value pairs for each element of the given array, where key is the element itself and value is provided by the [valueSelector] function applied to that key.
 * If any two elements are equal, the last one overwrites the former value in the map.
 * @sample samples.collections.Collections.Transformations.associateWithTo
 */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun <V, M : MutableMap<in Long, in V>> FloatArray.associateWithTo(destination: M, valueSelector: (Float) -> V): M {
    for (element in this) {
        destination.put(element, valueSelector(element))
    }
    return destination
}

/**
 * Populates and returns the [destination] mutable map with key-value pairs for each element of the given array, where key is the element itself and value is provided by the [valueSelector] function applied to that key.
 * If any two elements are equal, the last one overwrites the former value in the map.
 * @sample samples.collections.Collections.Transformations.associateWithTo
 */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun <V, M : MutableMap<in Float, in V>> DoubleArray.associateWithTo(destination: M, valueSelector: (Double) -> V): M {
    for (element in this) {
        destination.put(element, valueSelector(element))
    }
    return destination
}

/**
 * Populates and returns the [destination] mutable map with key-value pairs for each element of the given array, where key is the element itself and value is provided by the [valueSelector] function applied to that key.
 * If any two elements are equal, the last one overwrites the former value in the map.
 * @sample samples.collections.Collections.Transformations.associateWithTo
 */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun <V, M : MutableMap<in Double, in V>> BooleanArray.associateWithTo(destination: M, valueSelector: (Boolean) -> V): M {
    for (element in this) {
        destination.put(element, valueSelector(element))
    }
    return destination
}
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun <V, M : MutableMap<in Char, in V>>
CharArray.associateWithTo(destination: M, valueSelector: (Char) -> V): M {
    for (element in this) {
        destination.put(element, valueSelector(element))
    }
    return destination
}

/**
 * Appends all elements to the given [destination] collection.
 */
public fun <C : MutableCollection<in Byte>> ByteArray.toCollection(destination: C): C {
    for (item in this) {
        destination.add(item)
    }
    return destination
}

/**
 * Appends all elements to the given [destination] collection.
 */
public fun <T> Array<out T>.toCollection(destination: C): C {
    for (item in this) {
        destination.add(item)
    }
    return destination
}

/**
 * Appends all elements to the given [destination] collection.
 */
public fun <C : MutableCollection<in Short>> ShortArray.toCollection(destination: C): C {
    for (item in this) {
        destination.add(item)
    }
    return destination
}

/**
 * Appends all elements to the given [destination] collection.
 */
public fun <C : MutableCollection<in Int>> IntArray.toCollection(destination: C): C {
    for (item in this) {
        destination.add(item)
    }
    return destination
}

/**
 * Appends all elements to the given [destination] collection.
 */
public fun <C : MutableCollection<in Long>>
LongArray.toCollection(destination: C): C {
    for (item in this) {
        destination.add(item)
    }
    return destination
}

/**
 * Appends all elements to the given [destination] collection.
 */
public fun <C : MutableCollection<in Float>>
FloatArray.toCollection(destination: C): C {
    for (item in this) {
        destination.add(item)
    }
    return destination
}

/**
 * Appends all elements to the given [destination] collection.
 */
public fun <C : MutableCollection<in Double>>
DoubleArray.toCollection(destination: C): C {
    for (item in this) {
        destination.add(item)
    }
    return destination
}

/**
 * Appends all elements to the given [destination] collection.
 */
public fun <C : MutableCollection<in Boolean>>
BooleanArray.toCollection(destination: C): C {
    for (item in this) {
        destination.add(item)
    }
    return destination
}

/**
 * Appends all elements to the given [destination] collection.
 */
public fun <C : MutableCollection<in Char>>
CharArray.toCollection(destination: C): C {
    for (item in this) {
        destination.add(item)
    }
    return destination
}

/**
 * Returns a new [HashSet] of all elements.
 */
public fun <T> Array<out T>.toHashSet(): HashSet<T> {
    return toCollection(HashSet<T>(mapCapacity(size)))
}

/**
 * Returns a new [HashSet] of all elements.
 */
public fun ByteArray.toHashSet(): HashSet<Byte> {
    return toCollection(HashSet<Byte>(mapCapacity(size)))
}

/**
 * Returns a new [HashSet] of all elements.
 */
public fun ShortArray.toHashSet(): HashSet<Short> {
    return toCollection(HashSet<Short>(mapCapacity(size)))
}

/**
 * Returns a new [HashSet] of all elements.
 */
public fun IntArray.toHashSet(): HashSet<Int> {
    return toCollection(HashSet<Int>(mapCapacity(size)))
}

/**
 * Returns a new [HashSet] of all elements.
 */
public fun LongArray.toHashSet(): HashSet<Long> {
    return toCollection(HashSet<Long>(mapCapacity(size)))
}

/**
 * Returns a new [HashSet] of all elements.
 */
public fun FloatArray.toHashSet(): HashSet<Float> {
    return toCollection(HashSet<Float>(mapCapacity(size)))
}

/**
 * Returns a new [HashSet] of all elements.
 */
public fun DoubleArray.toHashSet(): HashSet<Double> {
    return toCollection(HashSet<Double>(mapCapacity(size)))
}

/**
 * Returns a new [HashSet] of all elements.
 */
public fun BooleanArray.toHashSet(): HashSet<Boolean> {
    return toCollection(HashSet<Boolean>(mapCapacity(size)))
}

/**
 * Returns a new [HashSet] of all elements.
 */
public fun CharArray.toHashSet(): HashSet<Char> {
    return toCollection(HashSet<Char>(mapCapacity(size,coerceAtMost(128))))
}

/**
 * Returns a new [HashSet] of all elements.
 */
public fun <T> Array<out T>.toList(): List<T> {
    return when (size) {
        0 -> emptyList()
        1 -> listOf(this[0])
        else -> this.toMutableList().toList()
    }
}

/**
 * Returns a new [List] containing all elements.
 */
public fun ByteArray.toList(): List<Byte> {
    return when (size) {
        0 -> emptyList()
        1 -> listOf(this[0])
        else -> this.toMutableList().toList()
    }
}

/**
 * Returns a new [List] containing all elements.
 */
public fun ShortArray.toList(): List<Short> {
    return when (size) {
        0 -> emptyList()
        1 -> listOf(this[0])
        else -> this.toMutableList().toList()
    }
}

/**
 * Returns a new [List] containing all elements.
 */
public fun IntArray.toList(): List<Int> {
    return when (size) {
        0 -> emptyList()
        1 -> listOf(this[0])
        else -> this.toMutableList().toList()
    }
}

/**
 * Returns a new [List] containing all elements.
 */
public fun LongArray.toList(): List<Long> {
    return when (size) {
        0 -> emptyList()
        1 -> listOf(this[0])
        else -> this.toMutableList().toList()
    }
}

/**
 * Returns a new [List] containing all elements.
 */
public fun FloatArray.toList(): List<Float> {
    return when (size) {
        0 -> emptyList()
        1 -> listOf(this[0])
        else -> this.toMutableList().toList()
    }
}

/**
 * Returns a new [List] containing all elements.
 */
public fun DoubleArray.toList(): List<Double> {
    return when (size) {
        0 -> emptyList()
        1 -> listOf(this[0])
        else -> this.toMutableList().toList()
    }
}

/**
 * Returns a new [List] containing all elements.
 */
public fun BooleanArray.toList(): List<Boolean> {
    return when (size) {
        0 -> emptyList()
        1 -> listOf(this[0])
        else -> this.toMutableList().toList()
    }
}

/**
 * Returns a new [List] containing all elements.
 */
public fun CharArray.toList(): List<Char> {
    return when (size) {
        0 -> emptyList()
        1 -> listOf(this[0])
        else -> this.toMutableList().toList()
    }
}
```kotlin
/**
 * Returns a [List] containing all elements.
 */
public fun IntArray.toList(): List<Int> {
    return when (size) {
        0 -> emptyList()
        1 -> listOf(this[0])
        else -> this.toMutableList()
    }
}

/**
 * Returns a [List] containing all elements.
 */
public fun LongArray.toList(): List<Long> {
    return when (size) {
        0 -> emptyList()
        1 -> listOf(this[0])
        else -> this.toMutableList()
    }
}

/**
 * Returns a [List] containing all elements.
 */
public fun FloatArray.toList(): List<Float> {
    return when (size) {
        0 -> emptyList()
        1 -> listOf(this[0])
        else -> this.toMutableList()
    }
}

/**
 * Returns a [List] containing all elements.
 */
public fun DoubleArray.toList(): List<Double> {
    return when (size) {
        0 -> emptyList()
        1 -> listOf(this[0])
        else -> this.toMutableList()
    }
}

/**
 * Returns a [List] containing all elements.
 */
public fun BooleanArray.toList(): List<Boolean> {
    return when (size) {
        0 -> emptyList()
        1 -> listOf(this[0])
        else -> this.toMutableList()
    }
}

/**
 * Returns a [List] containing all elements.
 */
public fun CharArray.toList(): List<Char> {
    return when (size) {
        0 -> emptyList()
        1 -> listOf(this[0])
        else -> this.toMutableList()
    }
}

/**
 * Returns a new [MutableList] filled with all elements of this array.
 */
public fun <T> Array<out T>.toMutableList(): MutableList<T> {
    return ArrayList(this.asCollection())
}

/**
 * Returns a new [MutableList] filled with all elements of this array.
 */
public fun ByteArray.toMutableList(): MutableList<Byte> {
    val list = ArrayList<Byte>(size)
    for (item in this) list.add(item)
    return list
}

/**
 * Returns a new [MutableList] filled with all elements of this array.
 */
public fun ShortArray.toMutableList(): MutableList<Short> {
    val list = ArrayList<Short>(size)
    for (item in this) list.add(item)
    return list
}

/**
 * Returns a new [MutableList] filled with all elements of this array.
 */
public fun IntArray.toMutableList(): MutableList<Int> {
    val list = ArrayList<Int>(size)
    for (item in this) list.add(item)
    return list
}

/**
 * Returns a new [MutableList] filled with all elements of this array.
 */
public fun LongArray.toMutableList(): MutableList<Long> {
    val list = ArrayList<Long>(size)
    for (item in this) list.add(item)
    return list
}

/**
 * Returns a new [MutableList] filled with all elements of this array.
 */
public fun FloatArray.toMutableList(): MutableList<Float> {
    val list = ArrayList<Float>(size)
    for (item in this) list.add(item)
    return list
}

/**
 * Returns a new [MutableList] filled with all elements of this array.
 */
public fun DoubleArray.toMutableList(): MutableList<Double> {
    val list = ArrayList<Double>(size)
    for (item in this) list.add(item)
    return list
}

/**
 * Returns a new [MutableList] filled with all elements of this array.
 */
public fun BooleanArray.toMutableList(): MutableList<Boolean> {
    val list = ArrayList<Boolean>(size)
    for (item in this) list.add(item)
    return list
}

/**
 * Returns a new [MutableList] filled with all elements of this array.
 */
public fun CharArray.toMutableList(): MutableList<Char> {
    val list = ArrayList<Char>(size)
    for (item in this) list.add(item)
    return list
}

/**
 * Returns a [Set] of all elements.
 * 
 * The returned set preserves the element iteration order of the original array.
 */
public fun <T> Array<out T>.toSet(): Set<T> {
    return when (size) {
        0 -> emptySet()
        1 -> setOf(this[0])
        else -> toCollection(LinkedHashSet<T>(mapCapacity(size)))
    }
}

/**
 * Returns a [Set] of all elements.
 * 
 * The returned set preserves the element iteration order of the original array.
 */
public fun ByteArray.toSet(): Set<Byte> {
    return when (size) {
        0 -> emptySet()
        1 -> setOf(this[0])
        else -> toCollection(LinkedHashSet<Byte>(mapCapacity(size)))
    }
}

/**
 * Returns a [Set] of all elements.
 * 
 * The returned set preserves the element iteration order of the original array.
 */
public fun ShortArray.toSet(): Set<Short> {
    return when (size) {
        0 -> emptySet()
        1 -> setOf(this[0])
        else -> toCollection(LinkedHashSet<Short>(mapCapacity(size)))
    }
}

/**
 * Returns a [Set] of all elements.
 * 
 * The returned set preserves the element iteration order of the original array.
 */
public fun IntArray.toSet(): Set<Int> {
    return when (size) {
        0 -> emptySet()
        1 -> setOf(this[0])
        else -> toCollection(LinkedHashSet<Int>(mapCapacity(size)))
    }
}

/**
 * Returns a [Set] of all elements.
 * 
 * The returned set preserves the element iteration order of the original array.
 */
public fun LongArray.toSet(): Set<Long> {
    return when (size) {
        0 -> emptySet()
        1 -> setOf(this[0])
        else -> toCollection(LinkedHashSet<Long>(mapCapacity(size)))
    }
}

/**
 * Returns a [Set] of all elements.
 * 
 * The returned set preserves the element iteration order of the original array.
 */
public fun FloatArray.toSet(): Set<Float> {
    return when (size) {
        0 -> emptySet()
        1 -> setOf(this[0])
        else -> toCollection(LinkedHashSet<Float>(mapCapacity(size)))
    }
}

/**
 * Returns a [Set] of all elements.
 * 
 * The returned set preserves the element iteration order of the original array.
 */
public fun DoubleArray.toSet(): Set<Double> {
    return when (size) {
        0 -> emptySet()
        1 -> setOf(this[0])
        else -> toCollection(LinkedHashSet<Double>(mapCapacity(size)))
    }
}

/**
 * Returns a [Set] of all elements.
 * 
 * The returned set preserves the element iteration order of the original array.
 */
public fun BooleanArray.toSet(): Set<Boolean> {
    return when (size) {
        0 -> emptySet()
        1 -> setOf(this[0])
        else -> toCollection(LinkedHashSet<Boolean>(mapCapacity(size)))
    }
}

/**
 * Returns a [Set] of all elements.
 * 
 * The returned set preserves the element iteration order of the original array.
 */
public fun CharArray.toSet(): Set<Char> {
    return when (size) {
        0 -> emptySet()
        1 -> setOf(this[0])
        else -> toCollection(LinkedHashSet<Char>(mapCapacity(size)))
    }
}
```

toCollection(LinkedHashSet<Float>(mapCapacity(size)))
    }
}  // Returns a [Set] of all elements.
}

/**
 * Returns a [Set] of all elements yielded from results of [transform] function being invoked on each element of original array.
 */
public fun DoubleArray.toSet(): Set<Double> {
    return when (size) {
        0 -> emptySet()
        1 -> setOf(this[0])
        else ->
            toCollection(LinkedHashSet<Double>(mapCapacity(size)))
    }
}  // Returns a [Set] of all elements.
}

/**
 * Returns a [Set] of all elements yielded from results of [transform] function being invoked on each element of original array.
 */
public fun CharArray.toSet(): Set<Char> {
    return when (size) {
        0 -> emptySet()
        1 -> setOf(this[0])
        else ->
            toCollection(LinkedHashSet<Char>(mapCapacity(size.coerceAtMost(128))))
    }
}  // Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@kotlin.jvm.JvmName("flatMapSequence")
public inline fun <T, R> Array<out T>.flatMap(transform: (T) -> Sequence<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}  // Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.
 */
public inline fun <R> ByteArray.flatMap(transform: (Byte) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}  // Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.
 */
public inline fun <R> ShortArray.flatMap(transform: (Short) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}  // Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.
 */
public inline fun <R> IntArray.flatMap(transform: (Int) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}  // Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.
 */
public inline fun <R> FloatArray.flatMap(transform: (Float) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}  // Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.
 */
public inline fun <R> DoubleArray.flatMap(transform: (Double) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}  // Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.
 */
public inline fun <R> BooleanArray.flatMap(transform: (Boolean) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}  // Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.
 */
public inline fun <R> CharArray.flatMap(transform: (Char) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}  // Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.

/*@SinceKotlin("1.4")*/
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("flatMapSequence")
public inline fun <T, R> Array<out T>.flatMap(transform: (T) -> Sequence<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}  // Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.

Open Source Used in security_automation identity-guest-cs 1240
invoked on each element\n and its index in the original array.\n * @sample samples.collections.Collections.Transformations.flatMapIndexed

*/
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("flatMapIndexedIterable")
@kotlin.internal.InlineOnly
@public inline fun <T, R> Array<out T>.flatMapIndexed(transform: (index: Int, T) -> Iterable<R>): List<R> {
    return flatMapIndexedTo(ArrayList<R>(), transform)
}

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each element\n and its index in the original array.\n * @sample samples.collections.Collections.Transformations.flatMapIndexed
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("flatMapIndexedIterable")
@kotlin.internal.InlineOnly
@public inline fun <R> ByteArray.flatMapIndexed(transform: (index: Int, Byte) -> Iterable<R>): List<R> {
    return flatMapIndexedTo(ArrayList<R>(), transform)
}

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each element\n and its index in the original array.\n * @sample samples.collections.Collections.Transformations.flatMapIndexed
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("flatMapIndexedIterable")
@kotlin.internal.InlineOnly
@public inline fun <R> ShortArray.flatMapIndexed(transform: (index: Int, Short) -> Iterable<R>): List<R> {
    return flatMapIndexedTo(ArrayList<R>(), transform)
}

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each element\n and its index in the original array.\n * @sample samples.collections.Collections.Transformations.flatMapIndexed
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("flatMapIndexedIterable")
@kotlin.internal.InlineOnly
@public inline fun <R> IntArray.flatMapIndexed(transform: (index: Int, Int) -> Iterable<R>): List<R> {
    return flatMapIndexedTo(ArrayList<R>(), transform)
}

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each element\n and its index in the original array.\n * @sample samples.collections.Collections.Transformations.flatMapIndexed
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("flatMapIndexedIterable")
@kotlin.internal.InlineOnly
@public inline fun <R> LongArray.flatMapIndexed(transform: (index: Int, Long) -> Iterable<R>): List<R> {
    return flatMapIndexedTo(ArrayList<R>(), transform)
}

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each element\n and its index in the original array.\n * @sample samples.collections.Collections.Transformations.flatMapIndexed
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("flatMapIndexedIterable")
@kotlin.internal.InlineOnly
@public inline fun <R> FloatArray.flatMapIndexed(transform: (index: Int, Float) -> Iterable<R>): List<R> {
    return flatMapIndexedTo(ArrayList<R>(), transform)
}

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each element\n and its index in the original array.\n * @sample samples.collections.Collections.Transformations.flatMapIndexed
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("flatMapIndexedIterable")
@kotlin.internal.InlineOnly
@public inline fun <R> DoubleArray.flatMapIndexed(transform: (index: Int, Double) -> Iterable<R>): List<R> {
    return flatMapIndexedTo(ArrayList<R>(), transform)
}

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each element\n and its index in the original array.\n * @sample samples.collections.Collections.Transformations.flatMapIndexed
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("flatMapIndexedIterable")
@kotlin.internal.InlineOnly
@public inline fun <R> BooleanArray.flatMapIndexed(transform: (index: Int, Boolean) -> Iterable<R>): List<R> {
    return flatMapIndexedTo(ArrayList<R>(), transform)
}
from results of [transform] function being invoked on each element and its index in the original array.

@sample samples.collections.Collections.Transformations.flatMapIndexed

*\@SinceKotlin("1.4")*\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\@kotlin.jvm.JvmName("flatMapIndexedIterable")\@kotlin.internal.InlineOnly\public inline fun <R> CharArray.flatMapIndexed(transform: (index: Int, Char) -> Iterable<R>): List<R> {\n    return flatMapIndexedTo(ArrayList<R>(), transform)\n}/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each element and its index in the original array.
 * @sample samples.collections.Collections.Transformations.flatMapIndexed
 */
@SinceKotlin("1.4")\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\@kotlin.jvm.JvmName("flatMapIndexedSequence")\@kotlin.internal.InlineOnly\public inline fun <T, R> Array<out T>.flatMapIndexed(transform: (index: Int, T) -> Sequence<R>): List<R> {\n    return flatMapIndexedTo(ArrayList<R>(), transform)\n}/**
 * Appends all elements yielded from results of [transform] function being invoked on each element and its index in the original array, to the given destination.
 */
@SinceKotlin("1.4")\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\@kotlin.jvm.JvmName("flatMapIndexedIterableTo")\@kotlin.internal.InlineOnly\public inline fun <T, R, C : MutableCollection<in R>> Array<out T>.flatMapIndexedTo(destination: C, transform: (index: Int, T) -> Iterable<R>): C {\n    var index = 0\n    for (element in this) {\n        val list = transform(index++, element)\n        destination.addAll(list)\n    }\n    return destination
}/**
 * Appends all elements yielded from results of [transform] function being invoked on each element and its index in the original array, to the given destination.
 */
@SinceKotlin("1.4")\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\@kotlin.jvm.JvmName("flatMapIndexedTo")\@kotlin.internal.InlineOnly\public inline fun <R, C : MutableCollection<in R>> ByteArray.flatMapIndexedTo(destination: C, transform: (index: Int, Byte) -> Iterable<R>): C {\n    var index = 0\n    for (element in this) {\n        val list = transform(index++, element)\n        destination.addAll(list)\n    }\n    return destination
}/**
 * Appends all elements yielded from results of [transform] function being invoked on each element and its index in the original array, to the given destination.
 */
@SinceKotlin("1.4")\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\@kotlin.jvm.JvmName("flatMapIndexedIterableTo")\@kotlin.internal.InlineOnly\public inline fun <R, C : MutableCollection<in R>> ShortArray.flatMapIndexedTo(destination: C, transform: (index: Int, Short) -> Iterable<R>): C {\n    var index = 0\n    for (element in this) {\n        val list = transform(index++, element)\n        destination.addAll(list)\n    }\n    return destination
}/**
 * Appends all elements yielded from results of [transform] function being invoked on each element and its index in the original array, to the given destination.
 */
@SinceKotlin("1.4")\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\@kotlin.jvm.JvmName("flatMapIndexedTo")\@kotlin.internal.InlineOnly\public inline fun <R, C : MutableCollection<in R>> IntArray.flatMapIndexedTo(destination: C, transform: (index: Int, Int) -> Iterable<R>): C {\n    var index = 0\n    for (element in this) {\n        val list = transform(index++, element)\n        destination.addAll(list)\n    }\n    return destination
}/**
 * Appends all elements yielded from results of [transform] function being invoked on each element and its index in the original array, to the given destination.
 */
@SinceKotlin("1.4")\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\@kotlin.jvm.JvmName("flatMapIndexedIterableTo")\@kotlin.internal.InlineOnly\public inline fun <R, C : MutableCollection<in R>> LongArray.flatMapIndexedTo(destination: C, transform: (index: Int, Long) -> Iterable<R>): C {\n    var index = 0\n    for (element in this) {\n        val list = transform(index++, element)\n        destination.addAll(list)\n    }\n    return destination
}
[destination].

```kotlin
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("flatMapIndexedIterableTo")
@kotlin.internal.InlineOnly
public inline fun <R, C : MutableCollection<in R>> FloatArray.flatMapIndexedTo(destination: C, transform: (index: Int, Float) -> Iterable<R>): C {
    var index = 0
    for (element in this) {
        val list = transform(index++, element)
        destination.addAll(list)
    }
    return destination
}

/**
 * Appends all elements yielded from results of [transform] function being invoked on each element in array, to the given [destination].
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("flatMapIndexedIterableTo")
@kotlin.internal.InlineOnly
public inline fun <R, C : MutableCollection<in R>> DoubleArray.flatMapIndexedTo(destination: C, transform: (index: Int, Double) -> Iterable<R>): C {
    var index = 0
    for (element in this) {
        val list = transform(index++, element)
        destination.addAll(list)
    }
    return destination
}

/**
 * Appends all elements yielded from results of [transform] function being invoked on each element in array, to the given [destination].
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("flatMapIndexedIterableTo")
@kotlin.internal.InlineOnly
public inline fun <R, C : MutableCollection<in R>> BooleanArray.flatMapIndexedTo(destination: C, transform: (index: Int, Boolean) -> Iterable<R>): C {
    var index = 0
    for (element in this) {
        val list = transform(index++, element)
        destination.addAll(list)
    }
    return destination
}

/**
 * Appends all elements yielded from results of [transform] function being invoked on each element in array, to the given [destination].
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("flatMapIndexedIterableTo")
@kotlin.internal.InlineOnly
public inline fun <T, R, C : MutableCollection<in R>> Array<out T>.flatMapIndexedTo(destination: C, transform: (index: Int, T) -> Sequence<R>): C {
    var index = 0
    for (element in this) {
        val list = transform(index++, element)
        destination.addAll(list)
    }
    return destination
}

/**
 * Appends all elements yielded from results of [transform] function being invoked on each element in array, to the given [destination].
 */
public inline fun <R, C : MutableCollection<in R>> ByteArray.flatMapTo(destination: C, transform: (Byte) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element)
        destination.addAll(list)
    }
    return destination
}

/**
 * Appends all elements yielded from results of [transform] function being invoked on each element in array, to the given [destination].
 */
public inline fun <R, C : MutableCollection<in R>> ShortArray.flatMapTo(destination: C, transform: (Short) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element)
        destination.addAll(list)
    }
    return destination
}

/**
 * Appends all elements yielded from results of [transform] function being invoked on each element in array, to the given [destination].
 */
public inline fun <R, C : MutableCollection<in R>> IntArray.flatMapTo(destination: C, transform: (Int) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element)
        destination.addAll(list)
    }
    return destination
}
```

Open Source Used in security_automation identity-guest-cs 1243
destination.addAll(list)  
}  
return destination\n} \n/**  
 * Appends all elements yielded from results of [transform] function being invoked on each element of original array, to the given [destination].\n */\npublic inline fun <R, C : MutableCollection\<in R>> LongArray.flatMapTo(destination: C, transform: (Long) -> Iterable\<R>): C \n{  
    for (element in this) {  
        val list = transform(element) \n        destination.addAll(list) \n    }  
    return destination\n} \n/**  
 * Appends all elements yielded from results of [transform] function being invoked on each element of original array, to the given [destination].\n */\npublic inline fun <R, C : MutableCollection\<in R>> FloatArray.flatMapTo(destination: C, transform: (Float) -> Iterable\<R>): C \n{  
    for (element in this) {  
        val list = transform(element) \n        destination.addAll(list) \n    }  
    return destination\n} \n/**  
 * Appends all elements yielded from results of [transform] function being invoked on each element of original array, to the given [destination].\n */\npublic inline fun <R, C : MutableCollection\<in R>> DoubleArray.flatMapTo(destination: C, transform: (Double) -> Iterable\<R>): C \n{  
    for (element in this) {  
        val list = transform(element) \n        destination.addAll(list) \n    }  
    return destination\n} \n/**  
 * Appends all elements yielded from results of [transform] function being invoked on each element of original array, to the given [destination].\n */\npublic inline fun <R, C : MutableCollection\<in R>> BooleanArray.flatMapTo(destination: C, transform: (Boolean) -> Iterable\<R>): C \n{  
    for (element in this) {  
        val list = transform(element) \n        destination.addAll(list) \n    }  
    return destination\n} \n/**  
 * Appends all elements yielded from results of [transform] function being invoked on each element of original array, to the given [destination].\n */\npublic inline fun <R, C : MutableCollection\<in R>> CharArray.flatMapTo(destination: C, transform: (Char) -> Iterable\<R>): C \n{  
    for (element in this) {  
        val list = transform(element) \n        destination.addAll(list) \n    }  
    return destination\n} \n/**  
 * Groups elements of the original array by the key returned by the given [keySelector] function\n * applied to each element and returns a map where each group key is associated with a list of  
 * corresponding elements.\n */\npublic inline fun <T, K> Array\<out T>.groupBy(keySelector: (T) -> K): Map\<K, List\<T>> \n{  
    return groupByTo(LinkedHashMap\<K, MutableList\<T>>(), keySelector)\n} \n/**  
 * Groups elements of the original array by the key returned by the given [keySelector] function\n * applied to each element and returns a map where each group key is associated with a list of  
 * corresponding elements.\n */\npublic inline fun <K> ByteArray.groupBy(keySelector: (Byte) -> K): Map\<K, List\<Byte>> \n{  
    return groupByTo(LinkedHashMap\<K, MutableList\<Byte>>(), keySelector)\n} \n/**  
 * Groups elements of the original array by the key returned by the given [keySelector] function\n * applied to each element and returns a map where each group key is associated with a list of  
 * corresponding elements.\n */\npublic inline fun <K> ShortArray.groupBy(keySelector: (Short) -> K): Map\<K, List\<Short>> \n{  
    return groupByTo(LinkedHashMap\<K, MutableList\<Short>>(), keySelector)\n} \n/**  
 * Groups elements of the original array by the key returned by the given [keySelector] function\n * applied to each element and returns a map where each group key is associated with a list of  
 * corresponding elements.\n */\npublic inline fun <K> IntArray.groupBy(keySelector: (Int) -> K): Map\<K, List\<Int>> \n{  
    return groupByTo(LinkedHashMap\<K, MutableList\<Int>>(), keySelector)\n}
given [keySelector] function applied to each element and returns a map where each group key is associated with a list of corresponding elements. The returned map preserves the entry iteration order of the keys produced from the original array. The returned map preserves the entry iteration order of the keys produced from the original array.

```
public fun <K> LongArray.groupBy(keySelector: (Long) -> K): Map<K, List<Long>> {
    return groupByTo(LinkedHashMap<K, MutableList<Long>>(), keySelector)
}
```

```
public fun <K> FloatArray.groupBy(keySelector: (Float) -> K): Map<K, List<Float>> {
    return groupByTo(LinkedHashMap<K, MutableList<Float>>(), keySelector)
}
```

```
public fun <K> DoubleArray.groupBy(keySelector: (Double) -> K): Map<K, List<Double>> {
    return groupByTo(LinkedHashMap<K, MutableList<Double>>(), keySelector)
}
```

```
public fun <K> BooleanArray.groupBy(keySelector: (Boolean) -> K): Map<K, List<Boolean>> {
    return groupByTo(LinkedHashMap<K, MutableList<Boolean>>(), keySelector)
}
```

```
public fun <K> CharArray.groupBy(keySelector: (Char) -> K): Map<K, List<Char>> {
    return groupByTo(LinkedHashMap<K, MutableList<Char>>(), keySelector)
}
```

```
public fun <T, K, V> Array<out T>.groupBy(keySelector: (T) -> K, valueTransform: (T) -> V): Map<K, List<V>> {
    return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)
}
```

```
public fun <K, V> ByteArray.groupBy(keySelector: (Byte) -> K, valueTransform: (Byte) -> V): Map<K, List<V>> {
    return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)
}
```

```
public fun <K, V> ShortArray.groupBy(keySelector: (Short) -> K, valueTransform: (Short) -> V): Map<K, List<V>> {
    return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)
}
```
returned by the [valueTransform] function applied to each element of the original array
* by the key returned by the given [keySelector] function applied to the element
* and returns a map where each group key is associated with a list of corresponding values.
* The returned map preserves the entry iteration order of the keys produced from the original array.

samples.collections.Collections.Transformations.groupByKeysAndValues

/**
 * Groups elements of the original array by the key returned by the given 
 * [keySelector] function and puts to the [destination] map each group key associated with
 * a list of corresponding elements.
 * @return The [destination] map.
 * @sample samples.collections.Collections.Transformations.groupBy
 */
public inline fun <T, K, M : MutableMap<in K, MutableList<T>>> Array<out T>.groupByTo(destination: M, keySelector: (T) -> K, valueTransform: (T) -> V): Map<K, List<V>> {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<T>() }
        list.add(element)
    }
    return destination
}
original array by the key returned by the given [keySelector] function \* applied to each element and puts to the [destination] map each group key associated with a list of corresponding elements.

```kotlin
public inline fun <K, M : MutableMap<in K, MutableList<Byte>>> ByteArray.groupByTo(destination: M, keySelector: (Byte) -> K): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<Byte>() }
        list.add(element)
    }
    return destination
}
```

```kotlin
public inline fun <K, M : MutableMap<in K, MutableList<Short>>> ShortArray.groupByTo(destination: M, keySelector: (Short) -> K): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<Short>() }
        list.add(element)
    }
    return destination
}
```

```kotlin
public inline fun <K, M : MutableMap<in K, MutableList<Int>>> IntArray.groupByTo(destination: M, keySelector: (Int) -> K): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<Int>() }
        list.add(element)
    }
    return destination
}
```

```kotlin
public inline fun <K, M : MutableMap<in K, MutableList<Long>>> LongArray.groupByTo(destination: M, keySelector: (Long) -> K): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<Long>() }
        list.add(element)
    }
    return destination
}
```

```kotlin
public inline fun <K, M : MutableMap<in K, MutableList<Float>>> FloatArray.groupByTo(destination: M, keySelector: (Float) -> K): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<Float>() }
        list.add(element)
    }
    return destination
}
```

```kotlin
public inline fun <K, M : MutableMap<in K, MutableList<Double>>> DoubleArray.groupByTo(destination: M, keySelector: (Double) -> K): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<Double>() }
        list.add(element)
    }
    return destination
}
```

```kotlin
public inline fun <K, M : MutableMap<in K, MutableList<Boolean>>> BooleanArray.groupByTo(destination: M, keySelector: (Boolean) -> K): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<Boolean>() }
        list.add(element)
    }
    return destination
}
```

```kotlin
public inline fun <K, M : MutableMap<in K, MutableList<Char>>> CharArray.groupByTo(destination: M, keySelector: (Char) -> K): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<Char>() }
```
list.add(element)
    }
    return destination
}

/**
 * Groups values returned by the [valueTransform] function applied to each element of the original array
 * by the key returned by the given [keySelector] function applied to the element
 * and puts to the [destination] map each group key associated with a list of corresponding values.
 * @return The [destination] map.
 */
public inline fun <T, K, V, M : MutableMap<in K, MutableList<V>>> Array<out T>.groupByTo(destination: M, keySelector: (T) -> K, valueTransform: (T) -> V): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<V>() }
        list.add(valueTransform(element))
    }
    return destination
}

/**
 * Groups values returned by the [valueTransform] function applied to each element of the original array
 * by the key returned by the given [keySelector] function applied to the element
 * and puts to the [destination] map each group key associated with a list of corresponding values.
 * @return The [destination] map.
 */
public inline fun <K, V, M : MutableMap<in K, MutableList<V>>> ByteArray.groupByTo(destination: M, keySelector: (Byte) -> K, valueTransform: (Byte) -> V): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<V>() }
        list.add(valueTransform(element))
    }
    return destination
}

/**
 * Groups values returned by the [valueTransform] function applied to each element of the original array
 * by the key returned by the given [keySelector] function applied to the element
 * and puts to the [destination] map each group key associated with a list of corresponding values.
 * @return The [destination] map.
 */
public inline fun <K, V, M : MutableMap<in K, MutableList<V>>> ShortArray.groupByTo(destination: M, keySelector: (Short) -> K, valueTransform: (Short) -> V): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<V>() }
        list.add(valueTransform(element))
    }
    return destination
}

/**
 * Groups values returned by the [valueTransform] function applied to each element of the original array
 * by the key returned by the given [keySelector] function applied to the element
 * and puts to the [destination] map each group key associated with a list of corresponding values.
 * @return The [destination] map.
 */
public inline fun <K, V, M : MutableMap<in K, MutableList<V>>> IntArray.groupByTo(destination: M, keySelector: (Int) -> K, valueTransform: (Int) -> V): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<V>() }
        list.add(valueTransform(element))
    }
    return destination
}

/**
 * Groups values returned by the [valueTransform] function applied to each element of the original array
 * by the key returned by the given [keySelector] function applied to the element
 * and puts to the [destination] map each group key associated with a list of corresponding values.
 * @return The [destination] map.
 */
public inline fun <K, V, M : MutableMap<in K, MutableList<V>>> LongArray.groupByTo(destination: M, keySelector: (Long) -> K, valueTransform: (Long) -> V): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<V>() }
        list.add(valueTransform(element))
    }
    return destination
}

/**
 * Groups values returned by the [valueTransform] function applied to each element of the original array
 * by the key returned by the given [keySelector] function applied to the element
 * and puts to the [destination] map each group key associated with a list of corresponding values.
 * @return The [destination] map.
 */
public inline fun <K, V, M : MutableMap<in K, MutableList<V>>> FloatArray.groupByTo(destination: M, keySelector: (Float) -> K, valueTransform: (Float) -> V): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<V>() }
        list.add(valueTransform(element))
    }
    return destination
}

/**
 * Groups values returned by the [valueTransform] function applied to each element of the original array
 * by the key returned by the given [keySelector] function applied to the element
 * and puts to the [destination] map each group key associated with a list of corresponding values.
 * @return The [destination] map.
 */
public inline fun <K, V, M : MutableMap<in K, MutableList<V>>> DoubleArray.groupByTo(destination: M, keySelector:
(Double) -> K, valueTransform: (Double) -> V): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<V>() }
        list.add(valueTransform(element))
    }
    return destination
}

/**
 * Groups values returned by the [valueTransform] function applied to each element of the original array
 * by the key returned by the given [keySelector] function applied to the element
 * and puts to the [destination] map each group key associated with a list of corresponding values.
 */
public inline fun <K, V, M : MutableMap<in K, MutableList<V>>> BooleanArray.groupByTo(destination: M, keySelector: (Boolean) -> K, valueTransform: (Boolean) -> V): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<V>() }
        list.add(valueTransform(element))
    }
    return destination
}

/**
 * Groups values returned by the [valueTransform] function applied to each element of the original array
 * by the key returned by the given [keySelector] function applied to the element
 * and puts to the [destination] map each group key associated with a list of corresponding values.
 */
public inline fun <K, V, M : MutableMap<in K, MutableList<V>>> CharArray.groupByTo(destination: M, keySelector: (Char) -> K, valueTransform: (Char) -> V): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<V>() }
        list.add(valueTransform(element))
    }
    return destination
}

/**
 * Creates a [Grouping] source from an array to be used later with one of group-and-fold operations
 * using the specified [keySelector] function to extract a key from each element.
 */
@SinceKotlin("1.1")
public inline fun <T, K> Array<out T>.groupingBy(crossinline keySelector: (T) -> K): Grouping<T, K> {
    return object : Grouping<T, K> {
        override fun sourceIterator(): Iterator<T> = this@groupingBy.iterator()
        override fun keyOf(element: T): K = keySelector(element)
    }
}

/**
 * Returns a list containing the results of applying the given [transform] function
 * to each element in the original array.
 */
public inline fun <T, R> Array<out T>.map(transform: (T) -> R): List<R> {
    return mapTo(ArrayList<R>(size), transform)
}

/**
 * Returns a list containing the results of applying the given [transform] function
 * to each element in the original array.
 */
public inline fun <R> ByteArray.map(transform: (Byte) -> R): List<R> {
    return mapTo(ArrayList<R>(size), transform)
}

/**
 * Returns a list containing the results of applying the given [transform] function
 * to each element in the original array.
 */
public inline fun <R> ShortArray.map(transform: (Short) -> R): List<R> {
    return mapTo(ArrayList<R>(size), transform)
}

/**
 * Returns a list containing the results of applying the given [transform] function
 * to each element in the original array.
 */
public inline fun <R> IntArray.map(transform: (Int) -> R): List<R> {
    return mapTo(ArrayList<R>(size), transform)
}

/**
 * Returns a list containing the results of applying the given [transform] function
 * to each element in the original array.
 */
public inline fun <R> LongArray.map(transform: (Long) -> R): List<R> {
    return mapTo(ArrayList<R>(size), transform)
}

/**
 * Returns a list containing the results of applying the given [transform] function
 * to each element in the original array.
 */
public inline fun <R> FloatArray.map(transform: (Float) -> R): List<R> {
    return mapTo(ArrayList<R>(size), transform)
}

/**
 * Returns a list containing the results of applying the given [transform] function
 * to each element in the original array.
 */
public inline fun <R> DoubleArray.map(transform: (Double) -> R): List<R> {
    return mapTo(ArrayList<R>(size), transform)
}

/**
 * Returns a list containing the results of applying the given [transform] function
 * to each element in the original array.
 */
public inline fun <R> BooleanArray.map(transform: (Boolean) -> R): List<R> {
    return mapTo(ArrayList<R>(size), transform)
}
@sample samples.collections.Collections.Transformations.map

INLINE FUN <R> CharArray.map(transform: (Char) -> R): List<R> {
    return mapTo(ArrayList<R>(size), transform)
}

Returns a list containing the results of applying the given [transform] function to each element and its index in the original array.

@param [transform] function that takes the index of an element and the element itself
and returns the result of the transform applied to the element.

@sample samples.collections.Collections.Transformations.map

INLINE FUN <R> ByteArray.mapIndexed(transform: (index: Int, Byte) -> R): List<R> {
    return mapIndexedTo(ArrayList<R>(size), transform)
}

Returns a list containing the results of applying the given [transform] function to each element and its index in the original array.

@param [transform] function that takes the index of an element and the element itself
and returns the result of the transform applied to the element.

@sample samples.collections.Collections.Transformations.map

INLINE FUN <R> ShortArray.mapIndexed(transform: (index: Int, Short) -> R): List<R> {
    return mapIndexedTo(ArrayList<R>(size), transform)
}

Returns a list containing the results of applying the given [transform] function to each element and its index in the original array.

@param [transform] function that takes the index of an element and the element itself
and returns the result of the transform applied to the element.

@sample samples.collections.Collections.Transformations.map

INLINE FUN <R> IntArray.mapIndexed(transform: (index: Int, Int) -> R): List<R> {
    return mapIndexedTo(ArrayList<R>(size), transform)
}

Returns a list containing the results of applying the given [transform] function to each element and its index in the original array.

@param [transform] function that takes the index of an element and the element itself
and returns the result of the transform applied to the element.

@sample samples.collections.Collections.Transformations.map

INLINE FUN <R> LongArray.mapIndexed(transform: (index: Int, Long) -> R): List<R> {
    return mapIndexedTo(ArrayList<R>(size), transform)
}

Returns a list containing the results of applying the given [transform] function to each element and its index in the original array.

@param [transform] function that takes the index of an element and the element itself
and returns the result of the transform applied to the element.

@sample samples.collections.Collections.Transformations.map

INLINE FUN <R> FloatArray.mapIndexed(transform: (index: Int, Float) -> R): List<R> {
    return mapIndexedTo(ArrayList<R>(size), transform)
}

Returns a list containing the results of applying the given [transform] function to each element and its index in the original array.

@param [transform] function that takes the index of an element and the element itself
and returns the result of the transform applied to the element.

@sample samples.collections.Collections.Transformations.map

INLINE FUN <R> DoubleArray.mapIndexed(transform: (index: Int, Double) -> R): List<R> {
    return mapIndexedTo(ArrayList<R>(size), transform)
}

Returns a list containing the results of applying the given [transform] function to each element and its index in the original array.

@param [transform] function that takes the index of an element and the element itself
and returns the result of the transform applied to the element.

@sample samples.collections.Collections.Transformations.map

INLINE FUN <R> BooleanArray.mapIndexed(transform: (index: Int, Boolean) -> R): List<R> {
    return mapIndexedTo(ArrayList<R>(size), transform)
}

Returns a list containing the results of applying the given [transform] function to each element and its index in the original array.

@param [transform] function that takes the index of an element and the element itself
and returns the result of the transform applied to the element.

@sample samples.collections.Collections.Transformations.map

INLINE FUN <R> BooleanArray.mapIndexedNotNull(transform: (index: Int, Boolean) -> R?): List<R> {
    return mapIndexedNotNullTo(ArrayList<R>(), transform)
}

Returns a list containing the results of applying the given [transform] function and only non-null results to the element.

@param [transform] function that takes the index of an element and the element itself
and returns the result of the transform applied to the element.

@sample samples.collections.Collections.Transformations.map

INLINE FUN <T> .mapIndexedNotNull(transform: (Int, T) -> R?): List<R> {
    return mapIndexedNotNullTo(ArrayList<R>(), transform)
}

Applies the given [transform] function to each element and its index in the original array
and appends only the non-null results to the given [destination].

@param [transform] function that takes the index of an element and the element itself
and returns the result of the transform applied to the element.

@sample samples.collections.Collections.Transformations.map

INLINE FUN <T> .mapIndexedNotNull(transform: (Int, T) -> R?): List<R> {
    return mapIndexedNotNullTo(ArrayList<R>(), transform)
}

Applies the given [transform] function to each element and its index in the original array
and appends only the non-null results to the given [destination].

@param [transform] function that takes the index of an element and the element itself
and returns the result of the transform applied to the element.
```kotlin
@sample samples.collections.Collections.Transformations.mapNotNull

fun <T, R : Any> Array<out T>.mapNotNull(transform: (T) -> R?): List<R> {
    return mapNotNullTo(ArrayList<R>(), transform)
}

fun <T, R : Any, C : MutableCollection<in R>> Array<out T>.mapTo(destination: C?, transform: (T) -> R?): C {
    return mapTo(destination, transform)
}
```

The text appears to be a code snippet from a Kotlin programming language, dealing with collections and transformations. The code is part of a class that provides functions for mapping elements of collections, handling null values, and mapping elements to a specified destination. The functions are inline and utilize the Kotlin language's feature of lazy evaluation and immutable collections, ensuring that operations are performed efficiently and safely.
index of that element and the element itself.

```kotlin
open fun FloatArray.withIndex(): Iterable<IndexedValue<Float>> { return IndexingIterable { iterator() } }.n
```

```kotlin
public fun DoubleArray.withIndex(): Iterable<IndexedValue<Double>> {
    return IndexingIterable { iterator() }.n
```

```kotlin
public fun BooleanArray.withIndex(): Iterable<IndexedValue<Boolean>> {
    return IndexingIterable { iterator() }.n
```

```kotlin
public fun CharArray.withIndex(): Iterable<IndexedValue<Char>> {
    return IndexingIterable { iterator() }.n
```

```kotlin
public fun FloatArray.withIndex(): Iterable<IndexedValue<Float>> {
    return IndexingIterable { iterator() }.n
```

```kotlin
public fun DoubleArray.withIndex(): Iterable<IndexedValue<Double>> {
    return IndexingIterable { iterator() }.n
```

```kotlin
public fun BooleanArray.withIndex(): Iterable<IndexedValue<Boolean>> {
    return IndexingIterable { iterator() }.n
```

```kotlin
public fun CharArray.withIndex(): Iterable<IndexedValue<Char>> {
    return IndexingIterable { iterator() }.n
```

```kotlin
public fun FloatArray.distinct(): List<Float> {
    return this.toMutableSet().toList().n
```

```kotlin
public fun DoubleArray.distinct(): List<Double> {
    return this.toMutableSet().toList().n
```

```kotlin
public fun BooleanArray.distinct(): List<Boolean> {
    return this.toMutableSet().toList().n
```

```kotlin
public fun CharArray.distinct(): List<Char> {
    return this.toMutableSet().toList().n
```

```kotlin
public fun FloatArray.distinct(): List<Float> {
    return this.toMutableSet().toList().n
```

```kotlin
public fun DoubleArray.distinct(): List<Double> {
    return this.toMutableSet().toList().n
```

```kotlin
public fun BooleanArray.distinct(): List<Boolean> {
    return this.toMutableSet().toList().n
```

```kotlin
public fun CharArray.distinct(): List<Char> {
    return this.toMutableSet().toList().n
```

```kotlin
public fun FloatArray.distinct(): List<Float> {
    return this.toMutableSet().toList().n
```

```kotlin
public fun DoubleArray.distinct(): List<Double> {
    return this.toMutableSet().toList().n
```

```kotlin
public fun BooleanArray.distinct(): List<Boolean> {
    return this.toMutableSet().toList().n
```

```kotlin
public fun CharArray.distinct(): List<Char> {
    return this.toMutableSet().toList().n
```

```kotlin
public fun FloatArray.distinct(): List<Float> {
    return this.toMutableSet().toList().n
```

```kotlin
public fun DoubleArray.distinct(): List<Double> {
    return this.toMutableSet().toList().n
```

```kotlin
public fun BooleanArray.distinct(): List<Boolean> {
    return this.toMutableSet().toList().n
```

```kotlin
public fun CharArray.distinct(): List<Char> {
    return this.toMutableSet().toList().n
```
Returns a list containing only elements from the given array\n * having distinct keys returned by the given [selector] function.\n * The elements in the resulting list are in the same order as they were in the source array.\n * @sample samples.collections.Collections.Transformations.distinctAndDistinctBy\n *\n ByteArray.distinctBy(selector: (Byte) -> K): List<Byte> \n for (e in this) \n val key = selector(e)\n if (set.add(key))\n list.add(e)\n return list\n\n**@sample** Returns a list containing only elements from the given array\n * having distinct keys returned by the given [selector] function.\n * The elements in the resulting list are in the same order as they were in the source array.\n * @sample samples.collections.Collections.Transformations.distinctAndDistinctBy\n *\n ShortArray.distinctBy(selector: (Short) -> K): List<Short> \n for (e in this) \n val key = selector(e)\n if (set.add(key))\n list.add(e)\n return list\n\n**@sample** Returns a list containing only elements from the given array\n * having distinct keys returned by the given [selector] function.\n * The elements in the resulting list are in the same order as they were in the source array.\n * @sample samples.collections.Collections.Transformations.distinctAndDistinctBy\n *\n IntArray.distinctBy(selector: (Int) -> K): List<Int> \n for (e in this) \n val key = selector(e)\n if (set.add(key))\n list.add(e)\n return list\n\n**@sample** Returns a list containing only elements from the given array\n * having distinct keys returned by the given [selector] function.\n * The elements in the resulting list are in the same order as they were in the source array.\n * @sample samples.collections.Collections.Transformations.distinctAndDistinctBy\n *\n FloatArray.distinctBy(selector: (Float) -> K): List<Float> \n for (e in this) \n val key = selector(e)\n if (set.add(key))\n list.add(e)\n return list\n\n**@sample** Returns a list containing only elements from the given array\n * having distinct keys returned by the given [selector] function.\n * The elements in the resulting list are in the same order as they were in the source array.\n * @sample samples.collections.Collections.Transformations.distinctAndDistinctBy\n *\n DoubleArray.distinctBy(selector: (Double) -> K): List<Double> \n for (e in this) \n val key = selector(e)\n if (set.add(key))\n list.add(e)\n return list\n\n**@sample** Returns a list containing only elements from the given array\n * having distinct keys returned by the given [selector] function.\n * The elements in the resulting list are in the same order as they were in the source array.\n * @sample samples.collections.Collections.Transformations.distinctAndDistinctBy\n *\n BooleanArray.distinctBy(selector: (Boolean) -> K): List<Boolean> \n for (e in this) \n val key = selector(e)\n if (set.add(key))\n list.add(e)\n return list\n\n**@sample** Returns a list containing only elements from the given array\n * having distinct keys returned by the given [selector] function.\n * The elements in the resulting list are in the same order as they were in the source array.\n * @sample samples.collections.Collections.Transformations.distinctAndDistinctBy\n *\n CharArray.distinctBy(selector: (Char) -> K): List<Char> \n for (e in this) \n val key = selector(e)\n if (set.add(key))\n list.add(e)\n return list\n\n**@sample** Returns a set containing all elements that are contained by both this array and the specified collection.\n * To get a set
```
open infix fun <T> Array<out T>.intersect(other: Iterable<T>): Set<T> { val set = this.toMutableSet() set.retainAll(other) return set } /* * Returns a set containing all elements that are contained by both this array and the specified collection. */ *
   * The returned set preserves the element iteration order of the original array. *
   * To get a set containing all elements that are contained at least in one of these collections use [union]. *

open infix fun ByteArray.intersect(other: Iterable<Byte>): Set<Byte> { val set = this.toMutableSet() set.retainAll(other) return set } /* * Returns a set containing all elements that are contained by both this array and the specified collection. */ *
   * The returned set preserves the element iteration order of the original array. *
   * To get a set containing all elements that are contained at least in one of these collections use [union]. *

open infix fun ShortArray.intersect(other: Iterable<Short>): Set<Short> { val set = this.toMutableSet() set.retainAll(other) return set } /* * Returns a set containing all elements that are contained by both this array and the specified collection. */ *
   * The returned set preserves the element iteration order of the original array. *
   * To get a set containing all elements that are contained at least in one of these collections use [union]. *

open infix fun IntArray.intersect(other: Iterable<Int>): Set<Int> { val set = this.toMutableSet() set.retainAll(other) return set } /* * Returns a set containing all elements that are contained by both this array and the specified collection. */ *
   * The returned set preserves the element iteration order of the original array. *
   * To get a set containing all elements that are contained at least in one of these collections use [union]. *

open infix fun LongArray.intersect(other: Iterable<Long>): Set<Long> { val set = this.toMutableSet() set.retainAll(other) return set } /* * Returns a set containing all elements that are contained by both this array and the specified collection. */ *
   * The returned set preserves the element iteration order of the original array. *
   * To get a set containing all elements that are contained at least in one of these collections use [union]. *

open infix fun DoubleArray.intersect(other: Iterable<Double>): Set<Double> { val set = this.toMutableSet() set.retainAll(other) return set } /* * Returns a set containing all elements that are contained by both this array and the specified collection. */ *
   * The returned set preserves the element iteration order of the original array. *
   * To get a set containing all elements that are contained at least in one of these collections use [union]. *

open infix fun FloatArray.intersect(other: Iterable<Float>): Set<Float> { val set = this.toMutableSet() set.retainAll(other) return set } /* * Returns a set containing all elements that are contained by both this array and the specified collection. */ *
   * The returned set preserves the element iteration order of the original array. *
   * To get a set containing all elements that are contained at least in one of these collections use [union]. *

open infix fun ByteArray.intersect(other: Iterable<Byte>): Set<Byte> { val set = this.toMutableSet() set.retainAll(other) return set } /* * Returns a set containing all elements that are contained by both this array and the specified collection. */ *
   * The returned set preserves the element iteration order of the original array. *
   * To get a set containing all elements that are contained at least in one of these collections use [union]. *
```

elements that are contained by this array and not contained by the specified collection.

* The returned set preserves the element iteration order of the original array.

* Returns a set containing all elements that are contained by this array and not contained by the specified collection.

* The returned set preserves the element iteration order of the original array.

* Returns a set containing all elements that are contained by this array and not contained by the specified collection.

* The returned set preserves the element iteration order of the original array.

* Returns a set containing all elements that are contained by this array and not contained by the specified collection.

* The returned set preserves the element iteration order of the original array.

* Returns a new [MutableSet] containing all distinct elements from the given array.

* The returned set preserves the element iteration order of the original array.

* Returns a new [MutableSet] containing all distinct elements from the given array.

* The returned set preserves the element iteration order of the original array.

* Returns a new [MutableSet] containing all distinct elements from the given array.

* The returned set preserves the element iteration order of the original array.

* Returns a new [MutableSet] containing all distinct elements from the given array.

* The returned set preserves the element iteration order of the original array.

* Returns a new [MutableSet] containing all distinct elements from the given array.

* The returned set preserves the element iteration order of the original array.

* Returns a new [MutableSet] containing all distinct elements from the given array.

* The returned set preserves the element iteration order of the original array.

* Returns a new [MutableSet] containing all distinct elements from the given array.

* The returned set preserves the element iteration order of the original array.

* Returns a new [MutableSet] containing all distinct elements from the given array.

* The returned set preserves the element iteration order of the original array.

* Returns a new [MutableSet] containing all distinct elements from the given array.

* The returned set preserves the element iteration order of the original array.

* Returns a new [MutableSet] containing all distinct elements from the given array.

* The returned set preserves the element iteration order of the original array.

* Returns a new [MutableSet] containing all distinct elements from the given array.

* The returned set preserves the element iteration order of the original array.

* Returns a new [MutableSet] containing all distinct elements from the given array.

* The returned set preserves the element iteration order of the original array.
from both collections.

The returned set preserves the element iteration order of the original array. Those elements of the [other] collection that are unique are iterated in the end in the order of the [other] collection. To get a set containing all elements that are contained in both collections use intersect.

Returns a set containing all distinct elements from both collections. The returned set preserves the element iteration order of the original array. Those elements of the [other] collection that are unique are iterated in the end in the order of the [other] collection. To get a set containing all elements that are contained in both collections use intersect.

Returns a set containing all distinct elements from both collections.
false then return false

/**
 * Returns `true` if all elements match the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.all
 */
public inline fun LongArray.all(predicate: (Long) -> Boolean): Boolean {
    for (element in this) if (!predicate(element)) return false
    return true
}

/**
 * Returns `true` if all elements match the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.all
 */
public inline fun FloatArray.all(predicate: (Float) -> Boolean): Boolean {
    for (element in this) if (!predicate(element)) return false
    return true
}

/**
 * Returns `true` if all elements match the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.all
 */
public inline fun DoubleArray.all(predicate: (Double) -> Boolean): Boolean {
    for (element in this) if (!predicate(element)) return false
    return true
}

/**
 * Returns `true` if all elements match the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.all
 */
public inline fun BooleanArray.all(predicate: (Boolean) -> Boolean): Boolean {
    for (element in this) if (!predicate(element)) return false
    return true
}

/**
 * Returns `true` if all elements match the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.all
 */
public inline fun CharArray.all(predicate: (Char) -> Boolean): Boolean {
    for (element in this) if (!predicate(element)) return false
    return true
}

/**
 * Returns `true` if any element matches the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.anyWithPredicate
 */
public inline fun <T> Array<out T>.any(predicate: (T) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}

/**
 * Returns `true` if any element matches the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.anyWithPredicate
 */
public inline fun ByteArray.any(predicate: (Byte) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}

/**
 * Returns `true` if any element matches the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.anyWithPredicate
 */
public inline fun ShortArray.any(predicate: (Short) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}

/**
 * Returns `true` if any element matches the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.anyWithPredicate
 */
public inline fun IntArray.any(predicate: (Int) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}

/**
 * Returns `true` if any element matches the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.anyWithPredicate
 */
public inline fun LongArray.any(predicate: (Long) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}

/**
 * Returns `true` if any element matches the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.anyWithPredicate
 */
public inline fun FloatArray.any(predicate: (Float) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}

/**
 * Returns `true` if any element matches the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.anyWithPredicate
 */
public inline fun DoubleArray.any(predicate: (Double) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}

/**
 * Returns `true` if any element matches the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.anyWithPredicate
 */
public inline fun BooleanArray.any(predicate: (Boolean) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}

/**
 * Returns `true` if any element matches the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.anyWithPredicate
 */
public inline fun CharArray.any(predicate: (Char) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}
(Float) -> Boolean): Boolean { for (element in this) if (predicate(element)) return true; return false
};

unary operator(): Boolean {
    for (element in this) if (predicate(element)) return true;
    return false
};

/**
 * Returns `true` if at least one element matches the given [predicate].
 */
public inline fun DoubleArray.any(predicate: (Double) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true;
    return false
};

/**
 * Returns `true` if at least one element matches the given [predicate].
 */
public inline fun BooleanArray.any(predicate: (Boolean) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true;
    return false
};

/**
 * Returns `true` if at least one element matches the given [predicate].
 */
public inline fun CharArray.any(predicate: (Char) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true;
    return false
};

/**
 * Returns the number of elements in this array.
 */
@kotlin.internal.InlineOnly
public inline fun <T> Array<out T>.count(): Int {
    return size
};

/**
 * Returns the number of elements in this array.
 */
@kotlin.internal.InlineOnly
public inline fun ByteArray.count(): Int {
    return size
};

/**
 * Returns the number of elements in this array.
 */
@kotlin.internal.InlineOnly
public inline fun ShortArray.count(): Int {
    return size
};

/**
 * Returns the number of elements in this array.
 */
@kotlin.internal.InlineOnly
public inline fun IntArray.count(): Int {
    return size
};

/**
 * Returns the number of elements in this array.
 */
@kotlin.internal.InlineOnly
public inline fun LongArray.count(): Int {
    return size
};

/**
 * Returns the number of elements in this array.
 */
@kotlin.internal.InlineOnly
public inline fun FloatArray.count(): Int {
    return size
};

/**
 * Returns the number of elements in this array.
 */
@kotlin.internal.InlineOnly
public inline fun DoubleArray.count(): Int {
    return size
};

/**
 * Returns the number of elements in this array.
 */
@kotlin.internal.InlineOnly
public inline fun BooleanArray.count(): Int {
    return size
};

/**
 * Returns the number of elements in this array.
 */
@kotlin.internal.InlineOnly
public inline fun CharArray.count(): Int {
    return size
};

/**
 * Returns the number of elements matching the given [predicate].
 */
public inline fun <T> Array<out T>.count(predicate: (T) -> Boolean): Int {
    var count = 0;
    for (element in this) if (predicate(element)) ++count;
    return count
};

/**
 * Returns the number of elements matching the given [predicate].
 */
public inline fun ByteArray.count(predicate: (Byte) -> Boolean): Int {
    var count = 0;
    for (element in this) if (predicate(element)) ++count;
    return count
};

/**
 * Returns the number of elements matching the given [predicate].
 */
public inline fun ShortArray.count(predicate: (Short) -> Boolean): Int {
    var count = 0;
    for (element in this) if (predicate(element)) ++count;
    return count
};

/**
 * Returns the number of elements matching the given [predicate].
 */
public inline fun IntArray.count(predicate: (Int) -> Boolean): Int {
    var count = 0;
    for (element in this) if (predicate(element)) ++count;
    return count
};

/**
 * Returns the number of elements matching the given [predicate].
 */
public inline fun LongArray.count(predicate: (Long) -> Boolean): Int {
    var count = 0;
    for (element in this) if (predicate(element)) ++count;
    return count
};

/**
 * Returns the number of elements matching the given [predicate].
 */
public inline fun FloatArray.count(predicate: (Float) -> Boolean): Int {
    var count = 0;
    for (element in this) if (predicate(element)) ++count;
    return count
};

/**
 * Returns the number of elements matching the given [predicate].
 */
public inline fun DoubleArray.count(predicate: (Double) -> Boolean): Int {
    var count = 0;
    for (element in this) if (predicate(element)) ++count;
    return count
};

/**
 * Returns the number of elements matching the given [predicate].
 */
public inline fun BooleanArray.count(predicate: (Boolean) -> Boolean): Int {
    var count = 0;
    for (element in this) if (predicate(element)) ++count;
    return count
};

/**
 * Returns the number of elements matching the given [predicate].
 */
public inline fun CharArray.count(predicate: (Char) -> Boolean): Int {
    var count = 0;
    for (element in this) if (predicate(element)) ++count;
    return count
};

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right to current accumulator value and each element.
 */
public inline fun <T> Array<T>.fold(initial: T, operation: (acc: T, T) -> T): T {
    var accumulator = initial;
    for (element in this) accumulator = operation(accumulator, element);
    return accumulator
};

/**
 * Accumulates value starting
with [initial] value and applying [operation] from left to right
* to current accumulator value and each element.

* Returns the specified [initial] value if the array is empty.
* @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.
* @unpublic inline fun <R>

ByteArray.fold(initial: R, operation: (acc: R, Byte) -> R): R {
    \n    \n    var accumulator = initia
    \n    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right
* to current accumulator value and each element.
* Returns the specified [initial] value if the array is empty.
* @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.
* @unpublic inline fun <R>

ShortArray.fold(initial: R, operation: (acc: R, Short) -> R): R {
    \n    \n    var accumulator = initia
    \n    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right
* to current accumulator value and each element.
* Returns the specified [initial] value if the array is empty.
* @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.
* @unpublic inline fun <R>

IntArray.fold(initial: R, operation: (acc: R, Int) -> R): R {
    \n    \n    var accumulator = initia
    \n    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right
* to current accumulator value and each element.
* Returns the specified [initial] value if the array is empty.
* @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.
* @unpublic inline fun <R>

LongArray.fold(initial: R, operation: (acc: R, Long) -> R): R {
    \n    \n    var accumulator = initia
    \n    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right
* to current accumulator value and each element.
* Returns the specified [initial] value if the array is empty.
* @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.
* @unpublic inline fun <R>

FloatArray.fold(initial: R, operation: (acc: R, Float) -> R): R {
    \n    \n    var accumulator = initia
    \n    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right
* to current accumulator value and each element.
* Returns the specified [initial] value if the array is empty.
* @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.
* @unpublic inline fun <R>

DoubleArray.fold(initial: R, operation: (acc: R, Double) -> R): R {
    \n    \n    var accumulator = initia
    \n    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right
* to current accumulator value and each element.
* Returns the specified [initial] value if the array is empty.
* @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.
* @unpublic inline fun <R>

CharArray.fold(initial: R, operation: (acc: R, Char) -> R): R {
    \n    \n    var accumulator = initia
    \n    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right
* to current accumulator value and each element.
* Returns the specified [initial] value if the array is empty.
* @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.
* @unpublic inline fun <R>

BooleanArray.fold(initial: R, operation: (acc: R, Boolean) -> R): R {
    \n    \n    var accumulator = initia
    \n    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right
* to current accumulator value and each element.
* Returns the specified [initial] value if the array is empty.
* @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.
* @unpublic inline fun <T, R> Array<out T>.foldIndexed(initial: R, operation: (index: Int, acc: R, T) -> R): R {
    \n    var index = 0
    \n    var accumulator = initia
    \n    return accumulator
}
specified [initial] value if the array is empty.\n* \n* @param [operation] function that takes the index of an element, current accumulator value\n* and the element itself, and calculates the next accumulator value.\n*

\n*/\n
public inline fun <R> ShortArray.foldIndexed(initial: R, operation: (Int, acc: R, Short) -> R): R {
    var index = 0\n    var accumulator = initial\n    for (element in this) accumulator = operation(index++, accumulator, element)\n    return accumulator\n}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right* to current accumulator value and each element with its index in the original array.\n* \n* @param [operation] function that takes the index of an element, current accumulator value\n* and the element itself, and calculates the next accumulator value.\n*/
public inline fun <R> CharArray.foldIndexed(initial: R, operation: (Int, acc: R, Char) -> R): R {
    var index = 0\n    var accumulator = initial\n    for (element in this) accumulator = operation(index++, accumulator, element)\n    return accumulator\n}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right* to current accumulator value and each element with its index in the original array.\n* \n* @param [operation] function that takes the index of an element, current accumulator value\n* and the element itself, and calculates the next accumulator value.\n*/
public inline fun <R> BooleanArray.foldIndexed(initial: R, operation: (Int, acc: R, Boolean) -> R): R {
    var index = 0\n    var accumulator = initial\n    for (element in this) accumulator = operation(index++, accumulator, element)\n    return accumulator\n}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right* to current accumulator value and each element with its index in the original array.\n* \n* @param [operation] function that takes the index of an element, current accumulator value\n* and the element itself, and calculates the next accumulator value.\n*/
public inline fun <R> DoubleArray.foldIndexed(initial: R, operation: (Int, acc: R, Double) -> R): R {
    var index = 0\n    var accumulator = initial\n    for (element in this) accumulator = operation(index++, accumulator, element)\n    return accumulator\n}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right* to current accumulator value and each element with its index in the original array.\n* \n* @param [operation] function that takes the index of an element, current accumulator value\n* and the element itself, and calculates the next accumulator value.\n*/
public inline fun <R> ByteArray.foldIndexed(initial: R, operation: (Int, acc: R, Byte) -> R): R {
    var index = 0\n    var accumulator = initial\n    for (element in this) accumulator = operation(index++, accumulator, element)\n    return accumulator\n}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right* to current accumulator value and each element with its index in the original array.\n* \n* @param [operation] function that takes the index of an element, current accumulator value\n* and the element itself, and calculates the next accumulator value.\n*/
public inline fun <R> IntArray.foldIndexed(initial: R, operation: (Int, acc: R, Int) -> R): R {
    var index = 0\n    var accumulator = initial\n    for (element in this) accumulator = operation(index++, accumulator, element)\n    return accumulator\n}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right* to current accumulator value and each element with its index in the original array.\n* \n* @param [operation] function that takes the index of an element, current accumulator value\n* and the element itself, and calculates the next accumulator value.\n*/
public inline fun <R> LongArray.foldIndexed(initial: R, operation: (Int, acc: R, Long) -> R): R {
    var index = 0\n    var accumulator = initial\n    for (element in this) accumulator = operation(index++, accumulator, element)\n    return accumulator\n}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right* to current accumulator value and each element with its index in the original array.\n* \n* @param [operation] function that takes the index of an element, current accumulator value\n* and the element itself, and calculates the next accumulator value.\n*/
public inline fun <R> FloatArray.foldIndexed(initial: R, operation: (Int, acc: R, Float) -> R): R {
    var index = 0\n    var accumulator = initial\n    for (element in this) accumulator = operation(index++, accumulator, element)\n    return accumulator\n}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right* to current accumulator value and each element with its index in the original array.\n* \n* @param [operation] function that takes the index of an element, current accumulator value\n* and the element itself, and calculates the next accumulator value.\n*/
public inline fun <R> BooleanArray.foldIndexed(initial: R, operation: (Int, acc: R, Boolean) -> R): R {
    var index = 0\n    var accumulator = initial\n    for (element in this) accumulator = operation(index++, accumulator, element)\n    return accumulator\n}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right* to current accumulator value and each element with its index in the original array.\n* \n* @param [operation] function that takes the index of an element, current accumulator value\n* and the element itself, and calculates the next accumulator value.\n*/
public inline fun <R> CharArray.foldIndexed(initial: R, operation: (Int, acc: R, Char) -> R): R {
    var index = 0\n    var accumulator = initial\n    for (element in this) accumulator = operation(index++, accumulator, element)\n    return accumulator\n}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right* to current accumulator value and each element with its index in the original array.\n* \n* @param [operation] function that takes the index of an element, current accumulator value\n* and the element itself, and calculates the next accumulator value.\n*/
public inline fun <R> ByteArray.foldIndexed(initial: R, operation: (Int, acc: R, Byte) -> R): R {
    var index = 0\n    var accumulator = initial\n    for (element in this) accumulator = operation(index++, accumulator, element)\n    return accumulator\n}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right* to current accumulator value and each element with its index in the original array.\n* \n* @param [operation] function that takes the index of an element, current accumulator value\n* and the element itself, and calculates the next accumulator value.\n*/
public inline fun <R> IntArray.foldIndexed(initial: R, operation: (Int, acc: R, Int) -> R): R {
    var index = 0\n    var accumulator = initial\n    for (element in this) accumulator = operation(index++, accumulator, element)\n    return accumulator\n}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right* to current accumulator value and each element with its index in the original array.\n* \n* @param [operation] function that takes the index of an element, current accumulator value\n* and the element itself, and calculates the next accumulator value.\n*/
public inline fun <R> LongArray.foldIndexed(initial: R, operation: (Int, acc: R, Long) -> R): R {
    var index = 0\n    var accumulator = initial\n    for (element in this) accumulator = operation(index++, accumulator, element)\n    return accumulator\n}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right* to current accumulator value and each element with its index in the original array.\n* \n* @param [operation] function that takes the index of an element, current accumulator value\n* and the element itself, and calculates the next accumulator value.\n*/
public inline fun <R> FloatArray.foldIndexed(initial: R, operation: (Int, acc: R, Float) -> R): R {
    var index = 0\n    var accumulator = initial\n    for (element in this) accumulator = operation(index++, accumulator, element)\n    return accumulator\n}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right* to current accumulator value and each element with its index in the original array.\n* \n* @param [operation] function that takes the index of an element, current accumulator value\n* and the element itself, and calculates the next accumulator value.\n*/
public inline fun <R> DoubleArray.foldIndexed(initial: R, operation: (Int, acc: R, Double) -> R): R {
    var index = 0\n    var accumulator = initial\n    for (element in this) accumulator = operation(index++, accumulator, element)\n    return accumulator\n}
// From right to left
T. foldRight(initial: R, operation: (T, acc: R) -> R): R
    return accumulator

{ int i = currentIndex - 1
    while (i >= 0) {
        accumulator = operation(acc, get(i--))
    }
    return accumulator
}
Accumulates value starting with [initial] value and applying [operation] from right to left starting at each element with its index in the array and current accumulator value.\n
\* Returns the specified [initial] value if the array is empty.\n
\* \n
\* @param [operation] function that takes the index of an element, the element itself and current accumulator value, and calculates the next accumulator value.

\* &nbsp;inline fun <T, R> Array<out T>.foldRightIndexed(initial: R, operation: (index: Int, T, acc: R) -> R): R {
\n var index = lastIndex \n var accumulator = initial\n while (index &gt;= 0) {
\n accumulator = operation(index, get(index), accumulator)\n \n return accumulator\n \}\n return accumulator\n}

\* &nbsp;\n
\* Returns the specified [initial] value if the array is empty.\n
\* \n
\* @param [operation] function that takes the index of an element, the element itself and current accumulator value, and calculates the next accumulator value.

\* &nbsp;public inline fun &lt;R&gt; ShortArray.foldRightIndexed(initial: R, operation: (index: Int, Short, acc: R) -> R): R {
\n var index = lastIndex \n var accumulator = initial\n while (index &gt;= 0) {
\n accumulator = operation(index, get(index), accumulator)\n \n return accumulator\n \}\n return accumulator\n}

\* &nbsp;\n
\* Returns the specified [initial] value if the array is empty.\n
\* \n
\* @param [operation] function that takes the index of an element, the element itself and current accumulator value, and calculates the next accumulator value.

\* &nbsp;public inline fun &lt;R&gt; IntArray.foldRightIndexed(initial: R, operation: (index: Int, Int, acc: R) -> R): R {
\n var index = lastIndex \n var accumulator = initial\n while (index &gt;= 0) {
\n accumulator = operation(index, get(index), accumulator)\n \n return accumulator\n \}\n return accumulator\n}

\* &nbsp;\n
\* Returns the specified [initial] value if the array is empty.\n
\* \n
\* @param [operation] function that takes the index of an element, the element itself and current accumulator value, and calculates the next accumulator value.

\* &nbsp;public inline fun &lt;R&gt; FloatArray.foldRightIndexed(initial: R, operation: (index: Int, Float, acc: R) -> R): R {
\n var index = lastIndex \n var accumulator = initial\n while (index &gt;= 0) {
\n accumulator = operation(index, get(index), accumulator)\n \n return accumulator\n \}\n return accumulator\n}

\* &nbsp;\n
\* Returns the specified [initial] value if the array is empty.\n
\* \n
\* @param [operation] function that takes the index of an element, the element itself and current accumulator value, and calculates the next accumulator value.

\* &nbsp;public inline fun &lt;R&gt; DoubleArray.foldRightIndexed(initial: R, operation: (index: Int, Double, acc: R) -> R): R {
\n var index = lastIndex \n var accumulator = initial\n while (index &gt;= 0) {
\n accumulator = operation(index, get(index), accumulator)\n \n return accumulator\n \}\n return accumulator\n}

\* &nbsp;\n
\* Returns the specified [initial] value if the array is empty.\n
\* \n
\* @param [operation] function that takes the index of an element, the element itself and current accumulator value, and calculates the next accumulator value.

\* &nbsp;public inline fun &lt;R&gt; ByteArray.foldRightIndexed(initial: R, operation: (index: Int, Byte, acc: R) -> R): R {
\n var index = lastIndex \n var accumulator = initial\n while (index &gt;= 0) {
\n accumulator = operation(index, get(index), accumulator)\n \n return accumulator\n \}\n return accumulator\n}

\* &nbsp;\n
\* Returns the specified [initial] value if the array is empty.\n
\* \n
\* @param [operation] function that takes the index of an element, the element itself and current accumulator value, and calculates the next accumulator value.

\* &nbsp;
specified [initial] value if the array is empty.

*param [operation] function that takes the index of an element, the element itself
* and current accumulator value, and calculates the next accumulator value.

*/
public inline fun <R> BooleanArray.foldRightIndexed(initial: R, operation: (index: Int, Boolean, acc: R) -> R):
R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

/**
 * Accumulates
 * value starting with [initial] value and applying [operation] from right to left to each element with its index in the
 * original array and current accumulator value.
 * *param [operation] function that takes the index of an element, the element itself
 * and current accumulator value, and calculates the next accumulator value.
 */
public inline fun <R> CharArray.foldRightIndexed(initial: R, operation: (index: Int, Char, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

/**
 * Performs the given [action] on each element.
 */
public inline fun <T> Array<out T>.forEach(action: (T) -> Unit): Unit {
    for (element in this) action(element)
}

/**
 * Performs the given [action] on each element, providing sequential index with the
 * element.
 */
public inline fun ByteArray.forEach(action: (Byte) -> Unit): Unit {
    for (element in this) action(element)
}

/**
 * Performs the given [action] on each element, providing sequential index with the
 * element.
 */
public inline fun ShortArray.forEach(action: (Short) -> Unit): Unit {
    for (element in this) action(element)
}

/**
 * Performs the given [action] on each element, providing sequential index with the
 * element.
 */
public inline fun IntArray.forEach(action: (Int) -> Unit): Unit {
    for (element in this) action(element)
}

/**
 * Performs the given [action] on each element, providing sequential index with the
 * element.
 */
public inline fun LongArray.forEach(action: (Long) -> Unit): Unit {
    for (element in this) action(element)
}

/**
 * Performs the given [action] on each element, providing sequential index with the
 * element.
 */
public inline fun FloatArray.forEach(action: (Float) -> Unit): Unit {
    for (element in this) action(element)
}

/**
 * Performs the given [action] on each element, providing sequential index with the
 * element.
 */
public inline fun DoubleArray.forEach(action: (Double) -> Unit): Unit {
    for (element in this) action(element)
}

/**
 * Performs the given [action] on each element, providing sequential index with the
 * element.
 */
public inline fun BooleanArray.forEach(action: (Boolean) -> Unit): Unit {
    for (element in this) action(element)
}

/**
 * Performs the given [action] on each element, providing sequential index with the
 * element.
 */
public inline fun CharArray.forEach(action: (Char) -> Unit): Unit {
    for (element in this) action(element)
}
maxByOrNull(selector)\n}n\n@Deprecated("Use maxByOrNull instead.",
ReplaceWith("this.maxByOrNull(selector())")n@DeprecatedSinceKotlin(warningSince = "1.4")npublic inline fun <R : Comparable<R>> DoubleArray.maxBy(selector: (Double) -> R): Double? {
    return maxByOrNull(selector)
}n\n@Deprecated("Use maxByOrNull instead.",
ReplaceWith("this.maxByOrNull(selector())")n@DeprecatedSinceKotlin(warningSince = "1.4")npublic inline fun <R : Comparable<R>> BooleanArray.maxBy(selector: (Boolean) -> R): Boolean? {
    return maxByOrNull(selector)
}

ReplaceWith("this.maxByOrNull(selector())")n@DeprecatedSinceKotlin(warningSince = "1.4")npublic inline function <R : Comparable<R>> CharArray.maxBy(selector: (Char) -> R): Char? {
    return maxByOrNull(selector)
}

/**
 * Returns the first element yielding the largest value of the given function or `null` if there are no elements.
 */
@DeprecatedSinceKotlin(warningSince = "1.4")n\npublic inline fun <R : Comparable<R>> ByteArray.maxBy(selector: (Byte) -> R): Byte? {
    return maxByOrNull(selector)
}

/**
 * Returns the first element yielding the largest value of the given function or `null` if there are no elements.
 */
@DeprecatedSinceKotlin(warningSince = "1.4")n\npublic inline fun <R : Comparable<R>> ShortArray.maxByOrNull(selector: (Short) -> R): Short? {
    return maxByOrNull(selector)
}

/**
 * Returns the first element yielding the largest value of the given function or `null` if there are no elements.
 */
@DeprecatedSinceKotlin(warningSince = "1.4")n\npublic inline fun <R : Comparable<R>> IntArray.maxByOrNull(selector: (Int) -> R): Int? {
    return maxByOrNull(selector)
}

/**
 * Returns the first element yielding the largest value of the given function or `null` if there are no elements.
 */
@DeprecatedSinceKotlin(warningSince = "1.4")n\npublic inline fun <R : Comparable<R>> LongArray.maxByOrNull(selector: (Long) -> R): Long? {
    return maxByOrNull(selector)
}

/**
 * Returns the first element yielding the largest value of the given function or `null` if there are no elements.
 */
@DeprecatedSinceKotlin(warningSince = "1.4")n\npublic inline fun <R : Comparable<R>> FloatArray.maxByOrNull(selector: (Float) -> R): Float? {
    return maxByOrNull(selector)
}

/**
 * Returns the first element yielding the largest value of the given function or `null` if there are no elements.
 */
@DeprecatedSinceKotlin(warningSince = "1.4")n\npublic inline fun <R : Comparable<R>> DoubleArray.maxByOrNull(selector: (Double) -> R): Double? {
    return maxByOrNull(selector)
}
-> R): Double? { val isEmpty = this.isEmpty() val maxElem = if (isEmpty) null else this[0] val lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxVal = selector(maxElem)
    for (i in 1..lastIndex) { val e = this[i]
        val v = selector(e)
        if (maxVal < v) { maxElem = e
            maxVal = v
        }
    }
    return maxElem
}

/**
 * Returns the first element yielding the largest value of the given function or `null` if there are no elements.
 * @sample samples.collections.Collections.Aggregates.maxByOrNull
 */
@SinceKotlin("1.4")
public inline fun <R : Comparable<R>> BooleanArray.maxByOrNull(selector: (Boolean) -> R): Boolean? { val isEmpty = this.isEmpty() val maxElem = if (isEmpty) this[0] else this[this.lastIndex]
    if (lastIndex == 0) return maxElem
    var maxVal = selector(maxElem)
    for (i in 1..lastIndex) { val e = this[i]
        val v = selector(e)
        if (maxVal < v) { maxElem = e
            maxVal = v
        }
    }
    return maxElem
}

/**
 * Returns the largest value among all values produced by [selector] function applied to each element in the array.
 * @sample samples.collections.Collections.Aggregates.maxByOrNull
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@kotlin.internal.InlineOnly
public inline fun <T> Array<out T>.maxOf(selector: (T) -> Double): Double {
    if (isEmpty()) throw NoSuchElementException()
    var maxVal = selector(this[0])
    for (i in 1..lastIndex) { val v = selector(this[i])
        maxVal = maxOf(maxVal, v)
    }
    return maxVal
}
*\@SinceKotlin("1.4")\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\@kotlin.internal.InlineOnly\@public inline fun ShortArray.maxOf(selector: Short -> Float): Float {\n    if (isEmpty()) throw NoSuchElementException()\n    var maxValue = selector(this[0])\n    for (i in 1..lastIndex) {\n        val v = selector(this[i])\n        maxValue = maxOf(maxValue, v)\n    }\n    return maxValue\n}\n
*\@SinceKotlin("1.4")\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\@kotlin.internal.InlineOnly\@public inline fun IntArray.maxOf(selector: Int -> Float): Float {\n    if (isEmpty()) throw NoSuchElementException()\n    var maxValue = selector(this[0])\n    for (i in 1..lastIndex) {\n        val v = selector(this[i])\n        maxValue = maxOf(maxValue, v)\n    }\n    return maxValue\n}\n
*\@SinceKotlin("1.4")\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\@kotlin.internal.InlineOnly\@public inline fun LongArray.maxOf(selector: Long -> Float): Float {\n    if (isEmpty()) throw NoSuchElementException()\n    var maxValue = selector(this[0])\n    for (i in 1..lastIndex) {\n        val v = selector(this[i])\n        maxValue = maxOf(maxValue, v)\n    }\n    return maxValue\n}\n
*\@SinceKotlin("1.4")\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\@kotlin.internal.InlineOnly\@public inline fun FloatArray.maxOf(selector: Float -> Float): Float {\n    if (isEmpty()) throw NoSuchElementException()\n    var maxValue = selector(this[0])\n    for (i in 1..lastIndex) {\n        val v = selector(this[i])\n        maxValue = maxOf(maxValue, v)\n    }\n    return maxValue\n}\n
*\@SinceKotlin("1.4")\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\@kotlin.internal.InlineOnly\@public inline fun DoubleArray.maxOf(selector: Double -> Float): Float {\n    if (isEmpty()) throw NoSuchElementException()\n    var maxValue = selector(this[0])\n    for (i in 1..lastIndex) {\n        val v = selector(this[i])\n        maxValue = maxOf(maxValue, v)\n    }\n    return maxValue\n}\n
*\@SinceKotlin("1.4")\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\@kotlin.internal.InlineOnly\@public inline fun BooleanArray.maxOf(selector: Boolean -> Float): Float {\n    if (isEmpty()) throw NoSuchElementException()\n    var maxValue = selector(this[0])\n    for (i in 1..lastIndex) {\n        val v = selector(this[i])\n        maxValue = maxOf(maxValue, v)\n    }\n    return maxValue\n}\n
*\@SinceKotlin("1.4")\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\@kotlin.internal.InlineOnly\@public inline fun CharArray.maxOf(selector: Char -> Float): Float {\n    if (isEmpty()) throw NoSuchElementException()\n    var maxValue = selector(this[0])\n    for (i in 1..lastIndex) {\n        val v = selector(this[i])\n        maxValue = maxOf(maxValue, v)\n    }\n    return maxValue\n}
fun maxOf(selector: (T) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue
}

ByteArray.maxOf(selector: (Byte) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue
}

ShortArray.maxOf(selector: (Short) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue
}

IntArray.maxOf(selector: (Int) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue
}

LongArray.maxOf(selector: (Long) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue
}

FloatArray.maxOf(selector: (Float) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue
}

DoubleArray.maxOf(selector: (Double) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue
}
NoSuchElementException if the array is empty.

```kotlin
/*\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>>
BooleanArray.maxOf(selector: (Boolean) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue
}"\n\nNoSuchElementException if the array is empty.

/*\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>>
CharArray.maxOf(selector: (Char) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue
}"\n\nNoSuchElementException if the array is empty.

/*\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <T> Array<out T>.maxOfOrNull(selector: (T) -> Double): Double? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}"\n\nNoSuchElementException if the array is empty.

/*\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>>
ByteArray.maxOfOrNull(selector: (Byte) -> Double): Double? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}"\n\nNoSuchElementException if the array is empty.

/*\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>>
ShortArray.maxOfOrNull(selector: (Short) -> Double): Double? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}"\n\nNoSuchElementException if the array is empty.

/*\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>>
IntArray.maxOfOrNull(selector: (Int) -> Double): Double? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}"\n\nNoSuchElementException if the array is empty.

/*\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>>
LongArray.maxOfOrNull(selector: (Long) -> Double): Double? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}"\n\nNoSuchElementException if the array is empty.
```
or `null` if there are no elements.

If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.

```kotlin
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByReturnType<out T> @kotlin.internal.InlineOnly
public inline fun Array<out T>.maxOfOrNull(selector: (T) -> Float): Float? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}
```

Returns the largest value among all values produced by [selector] function's * applied to each element in the array or `null` if there are no elements.

If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.

```kotlin
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByReturnType<out T> @kotlin.internal.InlineOnly
public inline fun ByteArray.maxOfOrNull(selector: (Byte) -> Float): Float? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}
```

Returns the largest value among all values produced by [selector] function's * applied to each element in the array or `null` if there are no elements.

If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.

```kotlin
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByReturnType<out T> @kotlin.internal.InlineOnly
public inline fun CharArray.maxOfOrNull(selector: (Char) -> Double): Double? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}
```

Returns the largest value among all values produced by [selector] function's * applied to each element in the array or `null` if there are no elements.

If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.

```kotlin
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByReturnType<out T> @kotlin.internal.InlineOnly
public inline fun BooleanArray.maxOfOrNull(selector: (Boolean) -> Double): Double? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}
```

Returns the largest value among all values produced by [selector] function's * applied to each element in the array or `null` if there are no elements.

If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.

```kotlin
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByReturnType<out T> @kotlin.internal.InlineOnly
public inline fun DoubleArray.maxOfOrNull(selector: (Double) -> Double): Double? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}
```

Returns the largest value among all values produced by [selector] function's * applied to each element in the array or `null` if there are no elements.

If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.

```kotlin
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByReturnType<out T> @kotlin.internal.InlineOnly
public inline fun FloatArray.maxOfOrNull(selector: (Float) -> Double): Double? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}
```

Returns the largest value among all values produced by [selector] function's * applied to each element in the array or `null` if there are no elements.

If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
the largest value among all values produced by \([\text{selector}]\) function\(\ast\) applied to each element in the array or `null` if there are no elements.\(\ast\ast\)\(\ast\) If any of values produced by \([\text{selector}]\) function is `NaN`, the returned result is `NaN`. \(\ast\ast\)\(\ast\)\n
```kotlin
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun IntArray.maxOfOrNull(selector: (Int) -> Float): Float? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}
```

default, the returned result is `NaN`.\(\ast\ast\)\(\ast\)

```kotlin
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun LongArray.maxOfOrNull(selector: (Long) -> Float): Float? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}
```

```kotlin
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun FloatArray.maxOfOrNull(selector: (Float) -> Float): Float? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}
```

```kotlin
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun DoubleArray.maxOfOrNull(selector: (Double) -> Float): Float? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}
```

```kotlin
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun BooleanArray.maxOfOrNull(selector: (Boolean) -> Float): Float? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}
```

```kotlin
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun CharArray.maxOfOrNull(selector: (Char) -> Float): Float? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}
```

```kotlin
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <T, R : Comparable<R>> Array<out T>.maxOfOrNull(selector: (T) -> R): R? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
    }
```

```kotlin
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <T, R : Comparable<R>> Array<out T>.maxOfOrNull(selector: (T) -> R): R? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
    }
```

```kotlin
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <T, R : Comparable<R>> Array<out T>.maxOfOrNull(selector: (T) -> R): R? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
```
return maxValue
* Returns the largest value among all values produced by [selector] function
* applied to each element in the array or `null` if there are no elements.

*\@SinceKotlin("1.4.0")\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\@kotlin.internal.InlineOnly\@public inline fun <R : Comparable<R>>
ByteArray.maxOfOrNull(selector: (Byte) -> R): R? {\n    if (isEmpty()) return null\n    var maxValue = selector(this[0])\n    for (i in 1..lastIndex) {\n        val v = selector(this[i])\n        if (maxValue < v) {\n            maxValue = v\n        }\n    }\n    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector]
 * function\n* applied to each element in the array or `null` if there are no elements.

*\@SinceKotlin("1.4.0")\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\@kotlin.internal.InlineOnly\@public inline fun <R : Comparable<R>>
ShortArray.maxOfOrNull(selector: (Short) -> R): R? {\n    if (isEmpty()) return null\n    var maxValue = selector(this[0])\n    for (i in 1..lastIndex) {\n        val v = selector(this[i])\n        if (maxValue < v) {\n            maxValue = v\n        }\n    }\n    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector]
 * function\n* applied to each element in the array or `null` if there are no elements.

*\@SinceKotlin("1.4.0")\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\@kotlin.internal.InlineOnly\@public inline fun <R : Comparable<R>>
IntArray.maxOfOrNull(selector: (Int) -> R): R? {\n    if (isEmpty()) return null\n    var maxValue = selector(this[0])\n    for (i in 1..lastIndex) {\n        val v = selector(this[i])\n        if (maxValue < v) {\n            maxValue = v\n        }\n    }\n    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector]
 * function\n* applied to each element in the array or `null` if there are no elements.

*\@SinceKotlin("1.4.0")\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\@kotlin.internal.InlineOnly\@public inline fun <R : Comparable<R>>
LongArray.maxOfOrNull(selector: (Long) -> R): R? {\n    if (isEmpty()) return null\n    var maxValue = selector(this[0])\n    for (i in 1..lastIndex) {\n        val v = selector(this[i])\n        if (maxValue < v) {\n            maxValue = v\n        }\n    }\n    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector]
 * function\n* applied to each element in the array or `null` if there are no elements.

*\@SinceKotlin("1.4.0")\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\@kotlin.internal.InlineOnly\@public inline fun <R : Comparable<R>>
FloatArray.maxOfOrNull(selector: (Float) -> R): R? {\n    if (isEmpty()) return null\n    var maxValue = selector(this[0])\n    for (i in 1..lastIndex) {\n        val v = selector(this[i])\n        if (maxValue < v) {\n            maxValue = v\n        }\n    }\n    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector]
 * function\n* applied to each element in the array or `null` if there are no elements.

*\@SinceKotlin("1.4.0")\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\@kotlin.internal.InlineOnly\@public inline fun <R : Comparable<R>>
DoubleArray.maxOfOrNull(selector: (Double) -> R): R? {\n    if (isEmpty()) return null\n    var maxValue = selector(this[0])\n    for (i in 1..lastIndex) {\n        val v = selector(this[i])\n        if (maxValue < v) {\n            maxValue = v\n        }\n    }\n    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector]
 * function\n* applied to each element in the array or `null` if there are no elements.

*\@SinceKotlin("1.4.0")\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\@kotlin.internal.InlineOnly\@public inline fun <R : Comparable<R>>
BooleanArray.maxOfOrNull(selector: (Boolean) -> R): R? {\n    if (isEmpty()) return null\n    var maxValue = selector(this[0])\n    for (i in 1..lastIndex) {\n        val v = selector(this[i])\n        if (maxValue < v) {\n            maxValue = v\n        }\n    }\n    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector]
 * function\n* applied to each element in the array or `null` if there are no elements.

*\@SinceKotlin("1.4.0")\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\@kotlin.internal.InlineOnly\@public inline fun <R : Comparable<R>>
CharArray.maxOfOrNull(selector: (Char) -> R): R? {\n    if (isEmpty()) return null\n    var maxValue = selector(this[0])\n    for (i in 1..lastIndex) {\n        val v = selector(this[i])\n        if (maxValue < v) {\n            maxValue = v\n        }\n    }\n    return maxValue
}
maxValue = v
} return maxValue

/**
 * Returns the largest value according to the provided 
 * among all values produced by [selector] function applied to each element in the array.
 * @throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <T, R> Array<T>.maxOfWith(comparator: Comparator<T, R>, selector: (T) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(maxValue, v) < 0) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array.
 * @throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R> ByteArray.maxOfWith(comparator: Comparator<in R>, selector: (Byte) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(maxValue, v) < 0) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array.
 * @throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R> ShortArray.maxOfWith(comparator: Comparator<in R>, selector: (Short) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(maxValue, v) < 0) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array.
 * @throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R> IntArray.maxOfWith(comparator: Comparator<in R>, selector: (Int) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(maxValue, v) < 0) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array.
 * @throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R> LongArray.maxOfWith(comparator: Comparator<in R>, selector: (Long) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(maxValue, v) < 0) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array.
 * @throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R> FloatArray.maxOfWith(comparator: Comparator<in R>, selector: (Float) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(maxValue, v) < 0) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array.
 * @throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R> DoubleArray.maxOfWith(comparator:
Comparator<in R>, selector: (Double) -> R) { if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(maxValue, v) < 0) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value according to the provided [comparator] among all values produced by [selector]
 * function applied to each element in the array.
 * @throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R> BooleanArray.maxOfWith(comparator: Comparator<in R>, selector: (Boolean) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(maxValue, v) < 0) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value according to the provided [comparator] among all values produced by [selector]
 * function applied to each element in the array.
 * @throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R> CharArray.maxOfWith(comparator: Comparator<in R>, selector: (Char) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(maxValue, v) < 0) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value according to the provided [comparator] among all values produced by [selector]
 * function applied to each element in the array or `null` if there are no elements.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <T, R> Array<out T>.maxOfWithOrNull(comparator: Comparator<in R>, selector: (T) -> R): R? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(maxValue, v) < 0) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value according to the provided [comparator] among all values produced by [selector]
 * function applied to each element in the array or `null` if there are no elements.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R> ByteArray.maxOfWithOrNull(comparator: Comparator<in R>, selector: (Byte) -> R): R? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(maxValue, v) < 0) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value according to the provided [comparator] among all values produced by [selector]
 * function applied to each element in the array or `null` if there are no elements.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R> ShortArray.maxOfWithOrNull(comparator: Comparator<in R>, selector: (Short) -> R): R? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(maxValue, v) < 0) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value according to the provided [comparator] among all values produced by [selector]
 * function applied to each element in the array or `null` if there are no elements.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R> IntArray.maxOfWithOrNull(comparator: Comparator<in R>, selector: (Int) -> R): R? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(maxValue, v) < 0) {
            maxValue = v
        }
    }
    return maxValue
}

Open Source Used In security_automation identity-guest-cs 1276
ByLambdaReturnType
LongArray.maxOfWithOrNull(comparator: Comparator<in R>, selector: (Long) -> R): R? {\n    if (isEmpty()) \n        return null\n    var maxValue = selector(this[0])\n    for (i in 1..lastIndex) {\n        val v = selector(this[i])\n        if (comparator.compare(maxValue, v) < 0) \n            maxValue = v\n    }\n    return maxValue
} \n
Returns the largest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array or ‘null’ if there are no elements.

ByLambdaReturnType
FloatArray.maxOfWithOrNull(comparator: Comparator<in R>, selector: (Float) -> R): R? {\n    if (isEmpty()) \n        return null\n    var maxValue = selector(this[0])\n    for (i in 1..lastIndex) {\n        val v = selector(this[i])\n        if (comparator.compare(maxValue, v) < 0) \n            maxValue = v\n    }\n    return maxValue
} \n
Returns the largest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array or ‘null’ if there are no elements.

ByLambdaReturnType
DoubleArray.maxOfWithOrNull(comparator: Comparator<in R>, selector: (Double) -> R): R? {\n    if (isEmpty()) \n        return null\n    var maxValue = selector(this[0])\n    for (i in 1..lastIndex) {\n        val v = selector(this[i])\n        if (comparator.compare(maxValue, v) < 0) \n            maxValue = v\n    }\n    return maxValue
} \n
Returns the largest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array or ‘null’ if there are no elements.

ByLambdaReturnType
CharArray.maxOfWithOrNull(comparator: Comparator<in R>, selector: (Char) -> R): R? {\n    if (isEmpty()) \n        return null\n    var maxValue = selector(this[0])\n    for (i in 1..lastIndex) {\n        val v = selector(this[i])\n        if (comparator.compare(maxValue, v) < 0) \n            maxValue = v\n    }\n    return maxValue
} \n
Returns the largest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array or ‘null’ if there are no elements.

ByLambdaReturnType
ByteArray.maxOrNull(): Byte? {\n    if (isEmpty()) \n        return null\n    var max = this[0]\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        if (max < e) max = e\n    }\n    return max
} \n
Returns the largest element or ‘null’ if there are no elements.

ByLambdaReturnType
ShortArray.maxOrNull(): Short? {\n    if (isEmpty()) \n        return null\n    var max = this[0]\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        if (max < e) max = e\n    }\n    return max
} \n
Returns the largest element or ‘null’ if there are no elements.

ByLambdaReturnType
IntArray.maxOrNull(): Int? {\n    if (isEmpty()) \n        return null\n    var max = this[0]\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        if (max < e) max = e\n    }\n    return max
} \n
Returns the largest element or ‘null’ if there are no elements.
```kotlin
return null
val max = this[0]
for (i in 1..lastIndex) {
    val e = this[i]
    if (max < e) max = e
}
for (i in 1..lastIndex) {
    val e = this[i]
    if (max < e) max = e
}
```

```
/**
 * Returns the largest element or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public fun LongArray.maxOrNull(): Long? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (max < e) max = e
    }
    return max
}

/**
 * Returns the largest element or `null` if there are no elements.
 * 
 * If any of elements is 'NaN' returns 'NaN'.
 */
@SinceKotlin("1.4")
public fun FloatArray.maxOrNull(): Float? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (max < e) max = e
    }
    return max
}

/**
 * Returns the largest element or `null` if there are no elements.
 * 
 * If any of elements is 'NaN' returns 'NaN'.
 */
@SinceKotlin("1.4")
public fun DoubleArray.maxOrNull(): Double? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (max < e) max = e
    }
    return max
}

/**
 * Returns the largest element or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public fun CharArray.maxOrNull(): Char? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (max < e) max = e
    }
    return max
}

@Deprecated("Use maxWithOrNull instead.",
    ReplaceWith("this.maxWithOrNull(comparator)"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun <T> Array<out T>.maxWith(comparator: Comparator<in T>): T? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(max, e) < 0) max = e
    }
    return max
}
```

```
Open Source Used In security_automation identity-guest-cs  1278
```
first element having the largest value according to the provided [comparator] or `null` if there are no elements.

```
@SinceKotlin("1.4")
public fun ShortArray.maxWithOrNull(comparator: Comparator<in Short>): Short? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(max, e) < 0) max = e
    }
    return max
}

@SinceKotlin("1.4")
public fun IntArray.maxWithOrNull(comparator: Comparator<in Int>): Int? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(max, e) < 0) max = e
    }
    return max
}

@SinceKotlin("1.4")
public fun LongArray.maxWithOrNull(comparator: Comparator<in Long>): Long? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(max, e) < 0) max = e
    }
    return max
}

@SinceKotlin("1.4")
public fun FloatArray.maxWithOrNull(comparator: Comparator<in Float>): Float? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(max, e) < 0) max = e
    }
    return max
}

@SinceKotlin("1.4")
public fun DoubleArray.maxWithOrNull(comparator: Comparator<in Double>): Double? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(max, e) < 0) max = e
    }
    return max
}

@SinceKotlin("1.4")
public fun BooleanArray.maxWithOrNull(comparator: Comparator<in Boolean>): Boolean? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(max, e) < 0) max = e
    }
    return max
}

@SinceKotlin("1.4")
public fun CharArray.maxWithOrNull(comparator: Comparator<in Char>): Char? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(max, e) < 0) max = e
    }
    return max
}
```

Open Source Used In security_automation identity-guest-cs 1279
ReplaceWith("this.minOrNull()")\n@DeprecatedSinceKotlin(warningSince = "1.4")\npublic fun CharArray.min(): Char? {\n    return minOrNull()\n}\n@Deprecated("Use minByOrNull instead.",\nReplaceWith("this.minByOrNull(selector())")\n@DeprecatedSinceKotlin(warningSince = "1.4")\npublic inline fun <T, R : Comparable<R>> Array<out T>.minBy(selector: (T) -> R): T? {\n    return minByOrNull(selector)\n}\n@Deprecated("Use minByOrNull instead.",\nReplaceWith("this.minByOrNull(selector())")\n@DeprecatedSinceKotlin(warningSince = "1.4")\npublic inline fun <R : Comparable<R>> ByteArray.minBy(selector: (Byte) -> R): Byte? {\n    return minByOrNull(selector)\n}\n@Deprecated("Use minByOrNull instead.",\nReplaceWith("this.minByOrNull(selector())")\n@DeprecatedSinceKotlin(warningSince = "1.4")\npublic inline fun <R : Comparable<R>> ShortArray.minBy(selector: (Short) -> R): Short? {\n    return minByOrNull(selector)\n}\n@Deprecated("Use minByOrNull instead.",\nReplaceWith("this.minByOrNull(selector())")\n@DeprecatedSinceKotlin(warningSince = "1.4")\npublic inline fun <R : Comparable<R>> IntArray.minBy(selector: (Int) -> R): Int? {\n    return minByOrNull(selector)\n}\n@Deprecated("Use minByOrNull instead.",\nReplaceWith("this.minByOrNull(selector())")\n@DeprecatedSinceKotlin(warningSince = "1.4")\npublic inline fun <R : Comparable<R>> LongArray.minBy(selector: (Long) -> R): Long? {\n    return minByOrNull(selector)\n}\n@Deprecated("Use minByOrNull instead.",\nReplaceWith("this.minByOrNull(selector())")\n@DeprecatedSinceKotlin(warningSince = "1.4")\npublic inline fun <R : Comparable<R>> FloatArray.minBy(selector: (Float) -> R): Float? {\n    return minByOrNull(selector)\n}\n@Deprecated("Use minByOrNull instead.",\nReplaceWith("this.minByOrNull(selector())")\n@DeprecatedSinceKotlin(warningSince = "1.4")\npublic inline fun <R : Comparable<R>> DoubleArray.minBy(selector: (Double) -> R): Double? {\n    return minByOrNull(selector)\n}\n@Deprecated("Use minByOrNull instead.",\nReplaceWith("this.minByOrNull(selector())")\n@DeprecatedSinceKotlin(warningSince = "1.4")\npublic inline fun <R : Comparable<R>> BooleanArray.minBy(selector: (Boolean) -> R): Boolean? {\n    return minByOrNull(selector)\n}\n@Deprecated("Use minByOrNull instead.",\nReplaceWith("this.minByOrNull(selector())")\n@DeprecatedSinceKotlin(warningSince = "1.4")\npublic inline fun <R : Comparable<R>> CharArray.minBy(selector: (Char) -> R): Char? {\n    return minByOrNull(selector)\n}\n/**\n * Returns the first element yielding the smallest value of the given function or 'null' if there are no elements.\n * @sample samples.collections.Collections.Aggregates.minByOrNull\n */\n@SinceKotlin("1.4")\npublic inline fun <T, R : Comparable<R>> Array<out T>.minByOrNull(selector: (T) -> R): T? {\n    if (isEmpty()) return null\n    var minElem = this[0]\n    val lastIndex = this.lastIndex\n    if (lastIndex == 0) return minElem\n    var minValue = selector(minElem)\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        val v = selector(e)\n        if (minValue > v) {\n            minElem = e\n            minValue = v\n        }\n    }\n    return minElem\n}\n/**\n * Returns the first element yielding the smallest value of the given function or 'null' if there are no elements.\n * @sample samples.collections.Collections.Aggregates.minByOrNull\n */\n@SinceKotlin("1.4")\npublic inline fun <R : Comparable<R>> ByteArray.minByOrNull(selector: (Byte) -> R): Byte? {\n    if (isEmpty()) return null\n    var minElem = this[0]\n    val lastIndex = this.lastIndex\n    if (lastIndex == 0) return minElem\n    var minValue = selector(minElem)\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        val v = selector(e)\n        if (minValue > v) {\n            minElem = e\n            minValue = v\n        }\n    }\n    return minElem\n}\n/**\n * Returns the first element yielding the smallest value of the given function or 'null' if there are no elements.\n * @sample samples.collections.Collections.Aggregates.minByOrNull\n */\n@SinceKotlin("1.4")\npublic inline fun <R : Comparable<R>> ShortArray.minByOrNull(selector: (Short) -> R): Short? {\n    if (isEmpty()) return null\n    var minElem = this[0]\n    val lastIndex = this.lastIndex\n    if (lastIndex == 0) return minElem\n    var minValue = selector(minElem)\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        val v = selector(e)\n        if (minValue > v) {\n            minElem = e\n            minValue = v\n        }\n    }\n    return minElem\n}\n/**\n * Returns the first element yielding the smallest value of the given function or 'null' if there are no elements.\n * @sample samples.collections.Collections.Aggregates.minByOrNull\n */\n@SinceKotlin("1.4")\npublic inline fun <R : Comparable<R>> IntArray.minByOrNull(selector: (Int) -> R): Int? {\n    if (isEmpty()) return null\n    var minElem = this[0]\n    val lastIndex = this.lastIndex\n    if (lastIndex == 0) return minElem\n    var minValue = selector(minElem)\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        val v = selector(e)\n        if (minValue > v) {\n            minElem = e\n            minValue = v\n        }\n    }\n    return minElem\n}\n/**\n * Returns the first element yielding the smallest value of the given function or 'null' if there are no elements.\n * @sample samples.collections.Collections.Aggregates.minByOrNull\n */\n@SinceKotlin("1.4")\npublic inline fun <R : Comparable<R>> LongArray.minByOrNull(selector: (Long) -> R): Long? {\n    if (isEmpty()) return null\n    var minElem = this[0]\n    val lastIndex = this.lastIndex\n    if (lastIndex == 0) return minElem\n    var minValue = selector(minElem)\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        val v = selector(e)\n        if (minValue > v) {\n            minElem = e\n            minValue = v\n        }\n    }\n    return minElem\n}\n/**\n * Returns the first element yielding the smallest value of the given function or 'null' if there are no elements.\n * @sample samples.collections.Collections.Aggregates.minByOrNull\n */\n@SinceKotlin("1.4")\npublic inline fun <R : Comparable<R>> FloatArray.minByOrNull(selector: (Float) -> R): Float? {\n    if (isEmpty()) return null\n    var minElem = this[0]\n    val lastIndex = this.lastIndex\n    if (lastIndex == 0) return minElem\n    var minValue = selector(minElem)\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        val v = selector(e)\n        if (minValue > v) {\n            minElem = e\n            minValue = v\n        }\n    }\n    return minElem\n}\n/**\n * Returns the first element yielding the smallest value of the given function or 'null' if there are no elements.\n * @sample samples.collections.Collections.Aggregates.minByOrNull\n */\n@SinceKotlin("1.4")\npublic inline fun <R : Comparable<R>> DoubleArray.minByOrNull(selector: (Double) -> R): Double? {\n    if (isEmpty()) return null\n    var minElem = this[0]\n    val lastIndex = this.lastIndex\n    if (lastIndex == 0) return minElem\n    var minValue = selector(minElem)\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        val v = selector(e)\n        if (minValue > v) {\n            minElem = e\n            minValue = v\n        }\n    }\n    return minElem\n}\n/**\n * Returns the first element yielding the smallest value of the given function or 'null' if there are no elements.\n * @sample samples.collections.Collections.Aggregates.minByOrNull\n */\n@SinceKotlin("1.4")\npublic inline fun <R : Comparable<R>> BooleanArray.minByOrNull(selector: (Boolean) -> R): Boolean? {\n    if (isEmpty()) return null\n    var minElem = this[0]\n    val lastIndex = this.lastIndex\n    if (lastIndex == 0) return minElem\n    var minValue = selector(minElem)\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        val v = selector(e)\n        if (minValue > v) {\n            minElem = e\n            minValue = v\n        }\n    }\n    return minElem\n}\n/**\n * Returns the first element yielding the smallest value of the given function or 'null' if there are no elements.\n * @sample samples.collections.Collections.Aggregates.minByOrNull\n */\n@SinceKotlin("1.4")\npublic inline fun <R : Comparable<R>> CharArray.minByOrNull(selector: (Char) -> R): Char? {\n    if (isEmpty()) return null\n    var minElem = this[0]\n    val lastIndex = this.lastIndex\n    if (lastIndex == 0) return minElem\n    var minValue = selector(minElem)\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        val v = selector(e)\n        if (minValue > v) {\n            minElem = e\n            minValue = v\n        }\n    }\n    return minElem\n}
@SinceKotlin("1.4")
public inline fun <R : Comparable<R>> IntArray.minByOrNull(selector: (Int) -> R): Int?
@SinceKotlin("1.4")
public inline fun <R : Comparable<R>> LongArray.minByOrNull(selector: (Long) -> R): Long?
@SinceKotlin("1.4")
public inline fun <R : Comparable<R>> FloatArray.minByOrNull(selector: (Float) -> R): Float?
@SinceKotlin("1.4")
public inline fun <R : Comparable<R>> DoubleArray.minByOrNull(selector: (Double) -> R): Double?
@SinceKotlin("1.4")
public inline fun <R : Comparable<R>> BooleanArray.minByOrNull(selector: (Boolean) -> R): Boolean?
@SinceKotlin("1.4")
public inline fun <R : Comparable<R>> CharArray.minByOrNull(selector: (Char) -> R): Char?
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@kotlin.internal.InlineOnly
public inline fun <T> Array<out T>.minOf(selector: (T) -> Double): Double

(i in 1..lastIndex) {\n  val v = selector(this[i])\n  minVal = minOf(minVal, v)\n}\nreturn

/*\n * @SinceKotlin("1.4")\n * @OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n * @OverloadResolution\n * @LambdaReturnType\n * public inline fun ShortArray.minOf(selector: (Short) -> Double): Double {
 *   if (isEmpty()) throw NoSuchElementException()\n *   var minVal = selector(this[0])\n *   for (i in 1..lastIndex) {\n *     val v = selector(this[i])\n *     minVal = minOf(minVal, v)\n *   }\n *   return
 * }

/*\n * Returns the smallest value among all values produced by [selector] function applied to each element in the array.\n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * @throws NoSuchElementException if the array is empty.\n * @SinceKotlin("1.4")\n * @OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n * @OverloadResolution\n * @LambdaReturnType\n * public inline fun IntArray.minOf(selector: (Int) -> Double): Double {
 *   if (isEmpty()) throw NoSuchElementException()\n *   var minVal = selector(this[0])\n *   for (i in 1..lastIndex) {\n *     val v = selector(this[i])\n *     minVal = minOf(minVal, v)\n *   }\n *   return
 * }

/*\n * Returns the smallest value among all values produced by [selector] function applied to each element in the array.\n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * @throws NoSuchElementException if the array is empty.\n * @SinceKotlin("1.4")\n * @OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n * @OverloadResolution\n * @LambdaReturnType\n * public inline fun LongArray.minOf(selector: (Long) -> Double): Double {
 *   if (isEmpty()) throw NoSuchElementException()\n *   var minVal = selector(this[0])\n *   for (i in 1..lastIndex) {\n *     val v = selector(this[i])\n *     minVal = minOf(minVal, v)\n *   }\n *   return
 * }

/*\n * Returns the smallest value among all values produced by [selector] function applied to each element in the array.\n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * @throws NoSuchElementException if the array is empty.\n * @SinceKotlin("1.4")\n * @OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n * @OverloadResolution\n * @LambdaReturnType\n * public inline fun FloatArray.minOf(selector: (Float) -> Double): Double {
 *   if (isEmpty()) throw NoSuchElementException()\n *   var minVal = selector(this[0])\n *   for (i in 1..lastIndex) {\n *     val v = selector(this[i])\n *     minVal = minOf(minVal, v)\n *   }\n *   return
 * }

/*\n * Returns the smallest value among all values produced by [selector] function applied to each element in the array.\n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * @throws NoSuchElementException if the array is empty.\n * @SinceKotlin("1.4")\n * @OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n * @OverloadResolution\n * @LambdaReturnType\n * public inline fun DoubleArray.minOf(selector: (Double) -> Double): Double {
 *   if (isEmpty()) throw NoSuchElementException()\n *   var minVal = selector(this[0])\n *   for (i in 1..lastIndex) {\n *     val v = selector(this[i])\n *     minVal = minOf(minVal, v)\n *   }\n *   return
 * }

/*\n * Returns the smallest value among all values produced by [selector] function applied to each element in the array.\n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * @throws NoSuchElementException if the array is empty.\n * @SinceKotlin("1.4")\n * @OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n * @OverloadResolution\n * @LambdaReturnType\n * public inline fun BooleanArray.minOf(selector: (Boolean) -> Boolean): Boolean {
 *   if (isEmpty()) throw NoSuchElementException()\n *   var minVal = selector(this[0])\n *   for (i in 1..lastIndex) {\n *     val v = selector(this[i])\n *     minVal = minOf(minVal, v)\n *   }\n *   return
 * }

/*\n * Returns the smallest value among all values produced by [selector] function applied to each element in the array.\n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * @throws NoSuchElementException if the array is empty.\n * @SinceKotlin("1.4")\n * @OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n * @OverloadResolution\n * @LambdaReturnType\n * public inline fun CharArray.minOf(selector: (Char) ->
Double): Double {\n    if (isEmpty()) throw NoSuchElementException();\n    var minValue = selector(this[0]);\n    for (i in 1..lastIndex) {\n        val v = selector(this[i]);\n        minValue = minOf(minValue, v);\n    }\n    return minValue;\n}/* * @return the smallest value among all values produced by [selector] function * applied to each element in the array.*/\n
Float): Float {\n    if (isEmpty()) throw NoSuchElementException();\n    var minValue = selector(this[0]);\n    for (i in 1..lastIndex) {\n        val v = selector(this[i]);\n        minValue = minOf(minValue, v);\n    }\n    return minValue;\n}/* * @return the smallest value among all values produced by [selector] function * applied to each element in the array.*/\n
ByteArray): Float {\n    if (isEmpty()) throw NoSuchElementException();\n    var minValue = selector(this[0]);\n    for (i in 1..lastIndex) {\n        val v = selector(this[i]);\n        minValue = minOf(minValue, v);\n    }\n    return minValue;\n}/* * @return the smallest value among all values produced by [selector] function * applied to each element in the array.*/\n
ShortArray): Float {\n    if (isEmpty()) throw NoSuchElementException();\n    var minValue = selector(this[0]);\n    for (i in 1..lastIndex) {\n        val v = selector(this[i]);\n        minValue = minOf(minValue, v);\n    }\n    return minValue;\n}/* * @return the smallest value among all values produced by [selector] function * applied to each element in the array.*/\n
IntArray): Float {\n    if (isEmpty()) throw NoSuchElementException();\n    var minValue = selector(this[0]);\n    for (i in 1..lastIndex) {\n        val v = selector(this[i]);\n        minValue = minOf(minValue, v);\n    }\n    return minValue;\n}/* * @return the smallest value among all values produced by [selector] function * applied to each element in the array.*/\n
LongArray): Float {\n    if (isEmpty()) throw NoSuchElementException();\n    var minValue = selector(this[0]);\n    for (i in 1..lastIndex) {\n        val v = selector(this[i]);\n        minValue = minOf(minValue, v);\n    }\n    return minValue;\n}/* * @return the smallest value among all values produced by [selector] function * applied to each element in the array.*/\n
FloatArray): Float {\n    if (isEmpty()) throw NoSuchElementException();\n    var minValue = selector(this[0]);\n    for (i in 1..lastIndex) {\n        val v = selector(this[i]);\n        minValue = minOf(minValue, v);\n    }\n    return minValue;\n}/* * @return the smallest value among all values produced by [selector] function * applied to each element in the array.*/
ByLambdaReturnType

@kotlin.internal.InlineOnly
public inline fun DoubleArray.minOf(selector: (Double) -> Float): Float {
    var minVal = selector(this[0])
    for (i in 1..lastIndex) {
        if (selector(this[i]) < minVal) {
            minVal = selector(this[i])
        }
    }
    return minVal
}

/**
 * Returns the smallest value among all values produced by [selector] function applied to each element in the array.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 * @throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType

@kotlin.internal.InlineOnly
public inline fun BooleanArray.minOf(selector: (Boolean) -> Float): Float {
    var minVal = selector(this[0])
    for (i in 1..lastIndex) {
        if (selector(this[i]) < minVal) {
            minVal = selector(this[i])
        }
    }
    return minVal
}

/**
 * Returns the smallest value among all values produced by [selector] function applied to each element in the array.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 * @throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType

@kotlin.internal.InlineOnly
public inline fun CharArray.minOf(selector: (Char) -> Float): Float {
    var minVal = selector(this[0])
    for (i in 1..lastIndex) {
        if (selector(this[i]) < minVal) {
            minVal = selector(this[i])
        }
    }
    return minVal
}

/**
 * Returns the smallest value among all values produced by [selector] function applied to each element in the array.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 * @throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType

@kotlin.internal.InlineOnly
public inline fun <T, R : Comparable<R>> Array<out T>.minOf(selector: (T) -> R): R {
    var minVal = selector(this[0])
    for (i in 1..lastIndex) {
        if (selector(this[i]) < minVal) {
            minVal = selector(this[i])
        }
    }
    return minVal
}

/**
 * Returns the smallest value among all values produced by [selector] function applied to each element in the array.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 * @throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType

@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> ByteArray.minOf(selector: (Byte) -> R): R {
    var minVal = selector(this[0])
    for (i in 1..lastIndex) {
        if (selector(this[i]) < minVal) {
            minVal = selector(this[i])
        }
    }
    return minVal
}

/**
 * Returns the smallest value among all values produced by [selector] function applied to each element in the array.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 * @throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType

@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> ShortArray.minOf(selector: (Short) -> R): R {
    var minVal = selector(this[0])
    for (i in 1..lastIndex) {
        if (selector(this[i]) < minVal) {
            minVal = selector(this[i])
        }
    }
    return minVal
}

/**
 * Returns the smallest value among all values produced by [selector] function applied to each element in the array.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 * @throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType

@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> IntArray.minOf(selector: (Int) -> R): R {
    var minVal = selector(this[0])
    for (i in 1..lastIndex) {
        if (selector(this[i]) < minVal) {
            minVal = selector(this[i])
        }
    }
    return minVal
}

/**
 * Returns the smallest value among all values produced by [selector] function applied to each element in the array.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 * @throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType

@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> LongArray.minOf(selector: (Long) -> R): R {
    var minVal = selector(this[0])
    for (i in 1..lastIndex) {
        if (selector(this[i]) < minVal) {
            minVal = selector(this[i])
        }
    }
    return minVal
}
ByLambdaReturnType\n@kotlin.internal.InlineOnly\n
public inline fun <R : Comparable<R>>
LongArray.minOf(selector: (Long) -> R): R
var
minValue = selector(this[0])
for (i in 1..lastIndex) {
    val v = selector(this[i])
    if (minValue > v) {
        minValue = v
    }
}
return minValue
*/

@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
FloatArray.minOf(selector: (Float) -> R): R
var
minValue = selector(this[0])
for (i in 1..lastIndex) {
    val v = selector(this[i])
    if (minValue > v) {
        minValue = v
    }
}
return minValue
*/

@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByteArray.minOfOrNull(selector: (Byte) -> Double): Double
var
minValue = selector(this[0])
for (i in 1..lastIndex) {
    val v = selector(this[i])
    minValue = minOf(minValue, v)
}
return minValue
*/

@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
CharArray.minOf(selector: (Char) -> R): R
var
minValue = selector(this[0])
for (i in 1..lastIndex) {
    val v = selector(this[i])
    if (minValue > v) {
        minValue = v
    }
}
return minValue
*/

@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByteArray.minOfOrNull(selector: (Byte) -> Double): Double
var
minValue = selector(this[0])
for (i in 1..lastIndex) {
    val v = selector(this[i])
    minValue = minOf(minValue, v)
}
return minValue
*/

@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByteArray.minOfOrNull(selector: (Byte) -> Double): Double
var
minValue = selector(this[0])
for (i in 1..lastIndex) {
    val v = selector(this[i])
    minValue = minOf(minValue, v)
}
return minValue
*/
public inline fun ShortArray.minOfOrNull(selector: (Short) -> Double): Double? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */
public inline fun IntArray.minOfOrNull(selector: (Int) -> Double): Double? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */
public inline fun LongArray.minOfOrNull(selector: (Long) -> Double): Double? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */
public inline fun FloatArray.minOfOrNull(selector: (Float) -> Double): Double? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */
public inline fun DoubleArray.minOfOrNull(selector: (Double) -> Double): Double? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */
public inline fun BooleanArray.minOfOrNull(selector: (Boolean) -> Double): Double? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */
public inline fun CharArray.minOfOrNull(selector: (Char) -> Double): Double? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}
Returns the smallest value among all values produced by [selector] function if there are no elements. If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.

Returns the smallest value among all values produced by [selector] function if there are no elements. If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.

Returns the smallest value among all values produced by [selector] function if there are no elements. If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.

Returns the smallest value among all values produced by [selector] function if there are no elements. If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.

Returns the smallest value among all values produced by [selector] function if there are no elements. If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.

Returns the smallest value among all values produced by [selector] function if there are no elements. If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.

Returns the smallest value among all values produced by [selector] function if there are no elements. If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.

Returns the smallest value among all values produced by [selector] function if there are no elements. If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
each element in the array or `null` if there are no elements.

If any of values produced by [selector] function
is `NaN`, the returned result is `NaN`.

```kotlin
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun BooleanArray.minOfOrNull(selector: (Boolean) -> Float): Float? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}
```

```kotlin
/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 *
 * If any of values produced by [selector] function
 * is `NaN`, the returned result is `NaN`.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun CharArray.minOfOrNull(selector: (Char) -> Float): Float? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}
```

```kotlin
/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 *
 * If any of values produced by [selector] function
 * is `NaN`, the returned result is `NaN`.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <T, R : Comparable<R>> Array<out T>.minOfOrNull(selector: (T) -> R): R? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}
```

```kotlin
/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 *
 * If any of values produced by [selector] function
 * is `NaN`, the returned result is `NaN`.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> ByteArray.minOfOrNull(selector: (Byte) -> R): R? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}
```

```kotlin
/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 *
 * If any of values produced by [selector] function
 * is `NaN`, the returned result is `NaN`.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> ShortArray.minOfOrNull(selector: (Short) -> R): R? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}
```

```kotlin
/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 *
 * If any of values produced by [selector] function
 * is `NaN`, the returned result is `NaN`.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> IntArray.minOfOrNull(selector: (Int) -> R): R? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}
```

```kotlin
/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 *
 * If any of values produced by [selector] function
 * is `NaN`, the returned result is `NaN`.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> LongArray.minOfOrNull(selector: (Long) -> R): R? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}
```

```kotlin
/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 *
 * If any of values produced by [selector] function
 * is `NaN`, the returned result is `NaN`.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> FloatArray.minOfOrNull(selector: (Float) -> R): R? {
    if (isEmpty()) return null
    var minValue =
fun DoubleArray.minOfOrNull(selector: (Double) -> R): R? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced
 * by [selector] function * applied to each element in the array or ‘null’ if there are no elements.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> DoubleArray.minOfOrNull(selector: (Double) -> R): R? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced
 * by [selector] function * applied to each element in the array or ‘null’ if there are no elements.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> BooleanArray.minOfOrNull(selector: (Boolean) -> R): R? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced
 * by [selector] function * applied to each element in the array or ‘null’ if there are no elements.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> CharArray.minOfOrNull(selector: (Char) -> R): R? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced
 * by [selector] function * applied to each element in the array or ‘null’ if there are no elements.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <T, R> Array<out T>.minOfWith(comparator: Comparator<in R>, selector: (T) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(minValue, v) > 0) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced
 * by [selector] function * applied to each element in the array or ‘null’ if there are no elements.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R> ByteArray.minOfWith(comparator: Comparator<in R>, selector: (Byte) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(minValue, v) > 0) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced
 * by [selector] function * applied to each element in the array or ‘null’ if there are no elements.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R> ShortArray.minOfWith(comparator: Comparator<in R>, selector: (Short) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(minValue, v) > 0) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced
 * by [selector] function * applied to each element in the array or ‘null’ if there are no elements.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R> IntArray.minOfWith(comparator: Comparator<in R>, selector: (Int) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(minValue, v) > 0) {
            minValue = v
        }
    }
    return minValue
}
Returns the smallest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array.

*\@SinceKotlin("1.4")*\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)*\@OverloadResolution

ByLambdaReturnType\@kotlin.internal.InlineOnly\@public inline fun <R> LongArray.minOfWith(comparator: Comparator<R>, selector: (Long) -> R): R { if (isEmpty()) throw NoSuchElementException() \n var minValue = selector(this[0]) \n for (i in 1..lastIndex) { \n \n \n return minValue
\n\n* Returns the smallest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array.

*\@SinceKotlin("1.4")*\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)*\@OverloadResolution

ByLambdaReturnType\@kotlin.internal.InlineOnly\@public inline fun <R> FloatArray.minOfWith(comparator: Comparator<R>, selector: (Float) -> R): R { if (isEmpty()) throw NoSuchElementException() \n var minValue = selector(this[0]) \n for (i in 1..lastIndex) { \n \n \n return minValue
\n\n* Returns the smallest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array.

*\@SinceKotlin("1.4")*\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)*\@OverloadResolution

ByLambdaReturnType\@kotlin.internal.InlineOnly\@public inline fun <R> DoubleArray.minOfWith(comparator: Comparator<R>, selector: (Double) -> R): R { if (isEmpty()) throw NoSuchElementException() \n var minValue = selector(this[0]) \n for (i in 1..lastIndex) { \n \n \n return minValue
\n\n* Returns the smallest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array.

*\@SinceKotlin("1.4")*\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)*\@OverloadResolution

ByLambdaReturnType\@kotlin.internal.InlineOnly\@public inline fun <R> BooleanArray.minOfWith(comparator: Comparator<R>, selector: (Boolean) -> R): R { if (isEmpty()) throw NoSuchElementException() \n var minValue = selector(this[0]) \n for (i in 1..lastIndex) { \n \n \n return minValue
\n\n* Returns the smallest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array.

*\@SinceKotlin("1.4")*\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)*\@OverloadResolution

ByLambdaReturnType\@kotlin.internal.InlineOnly\@public inline fun <R> CharArray.minOfWith(comparator: Comparator<R>, selector: (Char) -> R): R { if (isEmpty()) throw NoSuchElementException() \n var minValue = selector(this[0]) \n for (i in 1..lastIndex) { \n \n \n return minValue
\n\n* Returns the smallest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array.

*\@SinceKotlin("1.4")*\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)*\@OverloadResolution

ByLambdaReturnType\@kotlin.internal.InlineOnly\@public inline fun <T, R> Array<out T>.minOfWithOrNull(comparator: Comparator<in R>, selector: (T) -> R): R? { if (isEmpty()) return null \n var minValue = selector(this[0]) \n for (i in 1..lastIndex) { \n \n \n return minValue
\n\n* Returns the smallest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.

*\@SinceKotlin("1.4")*\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)*\@OverloadResolution

ByLambdaReturnType\@kotlin.internal.InlineOnly\@public inline fun <T, R> ByteArray.minOfWithOrNull(comparator: Comparator<in R>, selector: (Byte) -> R): R? { if (isEmpty()) return null \n var minValue = selector(this[0]) \n for (i in 1..lastIndex) { \n \n \n return minValue
\n\n* Returns the smallest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.

*\@SinceKotlin("1.4")*\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)*\@OverloadResolution

ByLambdaReturnType\@kotlin.internal.InlineOnly\@public inline fun <T, R> CharArray.minOfWithOrNull(comparator: Comparator<in R>, selector: (Char) -> R): R? { if (isEmpty()) return null \n var minValue = selector(this[0]) \n for (i in 1..lastIndex) { \n \n \n return minValue
\n\n* Returns the smallest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.

*\@SinceKotlin("1.4")*\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)*\@OverloadResolution

ByLambdaReturnType\@kotlin.internal.InlineOnly\@public inline fun <T, R> String.minOfWithOrNull(comparator: Comparator<in R>, selector: (String) -> R): R? { if (isEmpty()) return null \n var minValue = selector(this[0]) \n for (i in 1..lastIndex) { \n \n \n return minValue
\n\n* Returns the smallest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.
null
var minValue = selector(this[0])
for (i in 1..lastIndex) {
    val v = selector(this[i])
    if (comparator.compare(minValue, v) > 0) {
        minValue = v
    }
}
return minValue
}

/**
 * Returns the smallest value according to the provided [comparator] * among all values produced by [selector] function applied to each element in the array or 'null' if there are no elements.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R> ShortArray.minOfWithOrNull(comparator: Comparator<in R>, selector: (Short) -> R): R? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(minValue, v) > 0) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value according to the provided [comparator] * among all values produced by [selector] function applied to each element in the array or 'null' if there are no elements.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R> IntArray.minOfWithOrNull(comparator: Comparator<in R>, selector: (Int) -> R): R? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(minValue, v) > 0) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value according to the provided [comparator] * among all values produced by [selector] function applied to each element in the array or 'null' if there are no elements.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R> LongArray.minOfWithOrNull(comparator: Comparator<in R>, selector: (Long) -> R): R? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(minValue, v) > 0) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value according to the provided [comparator] * among all values produced by [selector] function applied to each element in the array or 'null' if there are no elements.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R> FloatArray.minOfWithOrNull(comparator: Comparator<in R>, selector: (Float) -> R): R? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(minValue, v) > 0) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value according to the provided [comparator] * among all values produced by [selector] function applied to each element in the array or 'null' if there are no elements.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R> DoubleArray.minOfWithOrNull(comparator: Comparator<in R>, selector: (Double) -> R): R? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(minValue, v) > 0) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value according to the provided [comparator] * among all values produced by [selector] function applied to each element in the array or 'null' if there are no elements.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R> BooleanArray.minOfWithOrNull(comparator: Comparator<in R>, selector: (Boolean) -> R): R? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(minValue, v) > 0) {
            minValue = v
        }
    }
    return minValue
}
CharArray.minOfWithOrNull(comparator: Comparator<in R>, selector: (Char) -> R): R? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(minValue, v) > 0) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest element or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public fun Array<out Double>.minOrNull(): Double? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (min > e) min = e
    }
    return min
}

/**
 * Returns the smallest element or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public fun Array<out Float>.minOrNull(): Float? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (min > e) min = e
    }
    return min
}

/**
 * Returns the smallest element or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public fun <T : Comparable<T>> Array<out T>.minOrNull(): T? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (min > e) min = e
    }
    return min
}

/**
 * Returns the smallest element or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public fun ByteArray.minOrNull(): Byte? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (min > e) min = e
    }
    return min
}

/**
 * Returns the smallest element or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public fun ShortArray.minOrNull(): Short? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (min > e) min = e
    }
    return min
}

/**
 * Returns the smallest element or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public fun IntArray.minOrNull(): Int? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (min > e) min = e
    }
    return min
}

/**
 * Returns the smallest element or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public fun LongArray.minOrNull(): Long? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (min > e) min = e
    }
    return min
}

/**
 * Returns the smallest element or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public fun FloatArray.minOrNull(): Float? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        min = minOf(min, e)
    }
    return min
}

/**
 * Returns the smallest element or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public fun DoubleArray.minOrNull(): Double? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        min = minOf(min, e)
    }
    return min
}

/**
 * Returns the smallest element or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public fun CharArray.minOrNull(): Char? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (min > e) min = e
    }
    return min
}

@Deprecated("Use minWithOrNull instead.",
ReplaceWith("this.minWithOrNull(comparator)"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun <T> Array<out T>.minWith(comparator: Comparator<in T>): T? {
    return this.minWithOrNull(comparator)
}

@Deprecated("Use minWithOrNull instead.",
ReplaceWith("this.minWithOrNull(comparator)"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun ByteArray.minWith(comparator: Comparator<in Byte>): Byte? {
    return this.minWithOrNull(comparator)
}

@Deprecated("Use minWithOrNull instead.",
ReplaceWith("this.minWithOrNull(comparator)"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun ShortArray.minWith(comparator: Comparator<in Short>): Short? {
    return this.minWithOrNull(comparator)
}

@Deprecated("Use minWithOrNull instead.",
ReplaceWith("this.minWithOrNull(comparator)"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun IntArray.minWith(comparator: Comparator<in Int>): Int? {
    return this.minWithOrNull(comparator)
}

@Deprecated("Use minWithOrNull instead.",
ReplaceWith("this.minWithOrNull(comparator)"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun LongArray.minWith(comparator: Comparator<in Long>): Long? {
    return this.minWithOrNull(comparator)
}
ReplaceWith("this.minWithOrNull(comparator)")\n@DeprecatedSinceKotlin(warningSince = "1.4")\n\npublic fun FloatArray.minWith(comparator: Comparator<in Float>): Float? {\n    return minWithOrNull(comparator)\n} \n\n@Deprecated(warningSince = "1.2")\n@DeprecatedSinceKotlin(warningSince = "1.4")\npublic fun DoubleArray.minWith(comparator: Comparator<in Double>): Double? {\n    return minWithOrNull(comparator)\n} \n\n@Deprecated(warningSince = "1.2")\n@DeprecatedSinceKotlin(warningSince = "1.4")\npublic fun BooleanArray.minWith(comparator: Comparator<in Boolean>): Boolean? {\n    return minWithOrNull(comparator)\n} \n\n@Deprecated(warningSince = "1.2")\n@DeprecatedSinceKotlin(warningSince = "1.4")\npublic fun CharArray.minWith(comparator: Comparator<in Char>): Char? {\n    return minWithOrNull(comparator)\n} \n\n/**\n * Returns the first element having the smallest value according to the provided [comparator] or `null` if there are no elements.\n */\n@SinceKotlin("1.4")\npublic fun <T> Array<out T>.minWithOrNull(comparator: Comparator<in T>): T? {\n    if (isEmpty()) return null\n    var min = this[0]\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        if (comparator.compare(min, e) > 0) min = e\n    }\n    return min\n} \n\n/**\n * Returns the first element having the smallest value according to the provided [comparator] or `null` if there are no elements.\n */\n@SinceKotlin("1.4")\npublic fun ByteArray.minWithOrNull(comparator: Comparator<in Byte>): Byte? {\n    if (isEmpty()) return null\n    var min = this[0]\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        if (comparator.compare(min, e) > 0) min = e\n    }\n    return min\n} \n\n/**\n * Returns the first element having the smallest value according to the provided [comparator] or `null` if there are no elements.\n */\n@SinceKotlin("1.4")\npublic fun ShortArray.minWithOrNull(comparator: Comparator<in Short>): Short? {\n    if (isEmpty()) return null\n    var min = this[0]\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        if (comparator.compare(min, e) > 0) min = e\n    }\n    return min\n} \n\n/**\n * Returns the first element having the smallest value according to the provided [comparator] or `null` if there are no elements.\n */\n@SinceKotlin("1.4")\npublic fun IntArray.minWithOrNull(comparator: Comparator<in Int>): Int? {\n    if (isEmpty()) return null\n    var min = this[0]\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        if (comparator.compare(min, e) > 0) min = e\n    }\n    return min\n} \n\n/**\n * Returns the first element having the smallest value according to the provided [comparator] or `null` if there are no elements.\n */\n@SinceKotlin("1.4")\npublic fun LongArray.minWithOrNull(comparator: Comparator<in Long>): Long? {\n    if (isEmpty()) return null\n    var min = this[0]\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        if (comparator.compare(min, e) > 0) min = e\n    }\n    return min\n} \n\n/**\n * Returns the first element having the smallest value according to the provided [comparator] or `null` if there are no elements.\n */\n@SinceKotlin("1.4")\npublic fun FloatArray.minWithOrNull(comparator: Comparator<in Float>): Float? {\n    if (isEmpty()) return null\n    var min = this[0]\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        if (comparator.compare(min, e) > 0) min = e\n    }\n    return min\n} \n\n/**\n * Returns the first element having the smallest value according to the provided [comparator] or `null` if there are no elements.\n */\n@SinceKotlin("1.4")\npublic fun DoubleArray.minWithOrNull(comparator: Comparator<in Double>): Double? {\n    if (isEmpty()) return null\n    var min = this[0]\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        if (comparator.compare(min, e) > 0) min = e\n    }\n    return min\n} \n\n/**\n * Returns the first element having the smallest value according to the provided [comparator] or `null` if there are no elements.\n */\n@SinceKotlin("1.4")\npublic fun BooleanArray.minWithOrNull(comparator: Comparator<in Boolean>): Boolean? {\n    if (isEmpty()) return null\n    var min = this[0]\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        if (comparator.compare(min, e) > 0) min = e\n    }\n    return min\n} \n\n/**\n * Returns `true` if the array has no elements.\n */\n@sample.samples.collections.Collections.Aggregates.none\n\n/**\n * Returns the first element having the smallest value according to the provided [comparator] or `null` if there are no elements.\n */\n@SinceKotlin("1.4")\npublic fun CharArray.minWithOrNull(comparator: Comparator<in Char>): Char? {\n    if (isEmpty()) return null\n    var min = this[0]\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        if (comparator.compare(min, e) > 0) min = e\n    }\n    return min\n}
T>.none(): Boolean {
    return isEmpty()
}

/**
 * Returns `true` if the array has no elements.
 * 
 * @sample samples.collections.Aggregates.none
 */
public fun ByteArray.none(): Boolean {
    return isEmpty()
}

/**
 * Returns `true` if the array has no elements.
 * 
 * @sample samples.collections.Aggregates.none
 */
public fun ShortArray.none(): Boolean {
    return isEmpty()
}

/**
 * Returns `true` if the array has no elements.
 * 
 * @sample samples.collections.Aggregates.none
 */
public fun IntArray.none(): Boolean {
    return isEmpty()
}

/**
 * Returns `true` if the array has no elements.
 * 
 * @sample samples.collections.Aggregates.none
 */
public fun LongArray.none(): Boolean {
    return isEmpty()
}

/**
 * Returns `true` if the array has no elements.
 * 
 * @sample samples.collections.Aggregates.none
 */
public fun FloatArray.none(): Boolean {
    return isEmpty()
}

/**
 * Returns `true` if the array has no elements.
 * 
 * @sample samples.collections.Aggregates.none
 */
public fun DoubleArray.none(): Boolean {
    return isEmpty()
}

/**
 * Returns `true` if no elements match the given 
 * [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.noneWithPredicate
 */
public inline fun <T> Array<out T>.none(predicate: (T) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return false
    return true
}

/**
 * Returns `true` if no elements match the given 
 * [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.noneWithPredicate
 */
public inline fun ByteArray.none(predicate: (Byte) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return false
    return true
}

/**
 * Returns `true` if no elements match the given 
 * [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.noneWithPredicate
 */
public inline fun ShortArray.none(predicate: (Short) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return false
    return true
}

/**
 * Returns `true` if no elements match the given 
 * [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.noneWithPredicate
 */
public inline fun IntArray.none(predicate: (Int) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return false
    return true
}

/**
 * Returns `true` if no elements match the given 
 * [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.noneWithPredicate
 */
public inline fun LongArray.none(predicate: (Long) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return false
    return true
}

/**
 * Returns `true` if no elements match the given 
 * [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.noneWithPredicate
 */
public inline fun FloatArray.none(predicate: (Float) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return false
    return true
}

/**
 * Returns `true` if no elements match the given 
 * [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.noneWithPredicate
 */
public inline fun DoubleArray.none(predicate: (Double) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return false
    return true
}

/**
 * Returns `true` if no elements match the given 
 * [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.noneWithPredicate
 */
public inline fun BooleanArray.none(predicate: (Boolean) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return false
    return true
}

/**
 * Returns `true` if no elements match the given 
 * [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.noneWithPredicate
 */
public inline fun CharArray.none(predicate: (Char) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return false
    return true
}

/**
 * Performs the given 
 * [action] on each element and returns the array itself afterwards.
 * 
 * @since Kotlin("1.4")
 */
@sinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun <T> Array<out T>.onEach(action: (T) -> Unit): Array<out T>

given [action] on each element, providing sequential index with the element.\n* and returns the array itself afterwards.\n* @param [action] function that takes the index of an element and the element itself\n* and performs the action on the element.\n*\n@SinceKotlin("1.4")\n@kotlin.internal.InlineOnly\npublic inline fun \nDoubleArray.onEachIndexed(action: (Int, Double) -> Unit): DoubleArray {\n    return apply {\n        forEachIndexed(action) \n    }\n}

/**
 * Performs the given [action] on each element, providing sequential index with the element,\n* and returns the array itself afterwards.\n* @param [action] function that takes the index of an element and the element itself\n* and performs the action on the element.\n*\n@SinceKotlin("1.4")\n@kotlin.internal.InlineOnly\npublic inline fun \nBooleanArray.onEachIndexed(action: (Int, Boolean) -> Unit): BooleanArray {\n    return apply {\n        forEachIndexed(action) \n    }\n}

/**
 * Performs the given [action] on each element, providing sequential index with the element,\n* and returns the array itself afterwards.\n* @param [action] function that takes the index of an element and the element itself\n* and performs the action on the element.\n*\n@SinceKotlin("1.4")\n@kotlin.internal.InlineOnly\npublic inline fun \nCharArray.onEachIndexed(action: (Int, Char) -> Unit): CharArray {\n    return apply {\n        forEachIndexed(action) \n    }\n}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right\n* to current accumulator value and each element.\n* @param [operation] function that takes current accumulator value and an element,\n* and calculates the next accumulator value.\n* @sample samples.collections.Collections.Aggregates.reduce\n */
public inline fun <S, T : S> Array<out T>.reduce(operation: (acc: S, T) -> S): S {
    if (isEmpty())
        throw 
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right\n* to current accumulator value and each element.\n* @param [operation] function that takes current accumulator value and an element,\n* and calculates the next accumulator value.\n* @sample samples.collections.Collections.Aggregates.reduce\n */
public inline fun ByteArray.reduce(operation: (acc: Byte, Byte) -> Byte): Byte {
    if (isEmpty())
        throw 
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right\n* to current accumulator value and each element.\n* @param [operation] function that takes current accumulator value and an element,\n* and calculates the next accumulator value.\n* @sample samples.collections.Collections.Aggregates.reduce\n */
public inline fun ShortArray.reduce(operation: (acc: Short, Short) -> Short): Short {
    if (isEmpty())
        throw 
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right\n* to current accumulator value and each element.\n* @param [operation] function that takes current accumulator value and an element,\n* and calculates the next accumulator value.\n* @sample samples.collections.Collections.Aggregates.reduce\n */
public inline fun IntArray.reduce(operation: (acc: Int, Int) -> Int): Int {
    if (isEmpty())
        throw 
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}
@param [operation] function that takes current accumulator value and an element,\n and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduce\n */

public inline fun LongArray.reduce(operation: (acc: Long, Long) -> Long): Long {
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right,\n * to current accumulator value and each element.\n * \n * @param [operation] function that takes current accumulator value and an element,\n * and calculates the next accumulator value and each element.\n * \n * @param [operation] \n * \n * @sample samples.collections.Collections.Aggregates.reduce\n */
public inline fun FloatArray.reduce(operation: (acc: Float, Float) -> Float): Float {
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right,\n * to current accumulator value and each element.\n * \n * @param [operation] function that takes current accumulator value and an element,\n * and calculates the next accumulator value and each element.\n * \n * @param [operation] \n * \n * @sample samples.collections.Collections.Aggregates.reduce\n */
public inline fun DoubleArray.reduce(operation: (acc: Double, Double) -> Double): Double {
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right,\n * to current accumulator value and each element.\n * \n * @param [operation] function that takes current accumulator value and an element,\n * and calculates the next accumulator value and each element.\n * \n * @param [operation] \n * \n * @sample samples.collections.Collections.Aggregates.reduce\n */
public inline fun BooleanArray.reduce(operation: (acc: Boolean, Boolean) -> Boolean): Boolean {
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right,\n * to current accumulator value and each element.\n * \n * @param [operation] function that takes the index of an element, current
 * accumulator value and the element itself,\n * and calculates the next accumulator value.\n * \n * @param [operation] \n * \n * @sample samples.collections.Collections.Aggregates.reduce\n */
public inline fun <S, T : S> Array<out T>.reduceIndexed(operation: (index: Int, acc: S, T) -> S): S {
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right,\n * to current accumulator value and each element with its index in the original array.\n * \n * @param [operation] function that takes the index of an element, current
 * accumulator value and the element itself,\n * and calculates the next accumulator value.\n * \n * @param [operation] \n * \n * @sample samples.collections.Collections.Aggregates.reduce\n */

Open Source Used In security_automation identity-guest-cs 1297
exception if this array is empty. If the array can be empty in an expected way,
* please use [reduceIndexedOrNull] instead. It returns `null` when its receiver is empty.

```kotlin
public inline fun ByteArray.reduceIndexed(operation: (index: Int, acc: Byte, Byte) -> Byte): Byte {
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}
```

Accumulates value starting with the first element and applying [operation] from left to right
* to current accumulator value and each element with its index in the original array.

```kotlin
public inline fun ShortArray.reduceIndexed(operation: (index: Int, acc: Short, Short) -> Short): Short {
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}
```

Accumulates value starting with the first element and applying [operation] from left to right
* to current accumulator value and each element with its index in the original array.

```kotlin
public inline fun IntArray.reduceIndexed(operation: (index: Int, acc: Int, Int) -> Int): Int {
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}
```

Accumulates value starting with the first element and applying [operation] from left to right
* to current accumulator value and each element with its index in the original array.

```kotlin
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}
```

Accumulates value starting with the first element and applying [operation] from left to right
* to current accumulator value and each element with its index in the original array.

```kotlin
public inline fun FloatArray.reduceIndexed(operation: (index: Int, acc: Float, Float) -> Float): Float {
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}
```

Accumulates value starting with the first element and applying [operation] from left to right
* to current accumulator value and each element with its index in the original array.

```kotlin
public inline fun * /public inline fun
```

ShortArray.reduceIndexed(operation: (index: Int, acc: Short, Short) -> Short): Short {
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}

Accumulates value starting with the first element and applying [operation] from left to right
* to current accumulator value and each element with its index in the original array.

```kotlin
public inline fun IntArray.reduceIndexed(operation: (index: Int, acc: Int, Int) -> Int): Int {
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}
```

Accumulates value starting with the first element and applying [operation] from left to right
* to current accumulator value and each element with its index in the original array.

```kotlin
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}
```

Accumulates value starting with the first element and applying [operation] from left to right
* to current accumulator value and each element with its index in the original array.

```kotlin
public inline fun FloatArray.reduceIndexed(operation: (index: Int, acc: Float, Float) -> Float): Float {
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}
```
@sample samples.collections.Collections.Aggregates.reduce in public inline fun DoubleArray.reduceIndexed(operation: (index: Int, acc: Double, Double) -> Double): Double {
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to
 * right in * to current accumulator value and each element with its index in the original array.
 * Throws an exception if this array is empty. If the array can be empty in an expected way,
 * please use [reduceIndexedOrNull] instead. It returns `null` when its receiver is empty.
 * @param [operation] function that takes the index of an element, current accumulator value and
 * the element itself, * and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduce
 */
public inline fun BooleanArray.reduceIndexed(operation: (index: Int, acc: Boolean, Boolean) -> Boolean): Boolean {
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to
 * right in * to current accumulator value and each element with its index in the original array.
 * Throws an exception if this array is empty. If the array can be empty in an expected way,
 * please use [reduceIndexedOrNull] instead. It returns `null` when its receiver is empty.
 * @param [operation] function that takes the index of an element, current accumulator value and
 * the element itself, * and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduce
 */
public inline fun CharArray.reduceIndexed(operation: (index: Int, acc: Char, Char) -> Char): Char {
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to
 * right in * to current accumulator value and each element with its index in the original array.
 * Returns `null` if the array is empty. If the array can be empty in an expected way,
 * please use [reduceIndexedOrNull] instead. It returns `null` when its receiver is empty.
 * @param [operation] function that takes the index of an element, current accumulator value and
 * the element itself, * and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduce
 */
public inline fun ByteArray.reduceIndexedOrNull(operation: (index: Int, acc: Byte, Byte) -> Byte): Byte? {
    if (isEmpty())
        return null
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to
 * right in * to current accumulator value and each element with its index in the original array.
 * Returns `null` if the array is empty. If the array can be empty in an expected way,
 * please use [reduceIndexedOrNull] instead. It returns `null` when its receiver is empty.
 * @param [operation] function that takes the index of an element, current accumulator value and
 * the element itself, * and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduce
 */
public inline fun ShortArray.reduceIndexedOrNull(operation: (index: Int, acc: Short, Short) -> Short): Short? {
    if (isEmpty())
        return null
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}
value.

@sample samples.collections.Collections.Aggregates.reduceOrNullable

*\n*

*\n@sample samples.collections.Collections.Aggregates.reduceOrNullable

/**
 * Returns `null` if the array is empty.
 * @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduceOrNullable
 */
public inline fun IntArray.reduceOrNull(operation: (index: Int, acc: Int, Int) -> Int): Int? {
    if (isEmpty())
        return null
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element with its index in the original array.
 * @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduceOrNullable
 */
@SinceKotlin("1.4")
    if (isEmpty())
        return null
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element with its index in the original array.
 * @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduceOrNullable
 */
@SinceKotlin("1.4")
public inline fun FloatArray.reduceIndexedOrNull(operation: (index: Int, acc: Float, Float) -> Float): Float? {
    if (isEmpty())
        return null
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element with its index in the original array.
 * @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduceOrNullable
 */
@SinceKotlin("1.4")
public inline fun DoubleArray.reduceIndexedOrNull(operation: (index: Int, acc: Double, Double) -> Double): Double? {
    if (isEmpty())
        return null
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element with its index in the original array.
 * @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduceOrNullable
 */
@SinceKotlin("1.4")
public inline fun BooleanArray.reduceIndexedOrNull(operation: (index: Int, acc: Boolean, Boolean) -> Boolean): Boolean? {
    if (isEmpty())
        return null
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element with its index in the original array.
 * @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduceOrNullable
 */
@SinceKotlin("1.4")
public inline fun CharArray.reduceIndexedOrNull(operation: (index: Int, acc: Char, Char) -> Char): Char? {
    if (isEmpty())
        return null
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element with its index in the original array.
 * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduceOrNullable
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public inline fun <S, T : S> Array<out T>.reduceOrNull(operation: (acc: T, S) -> S): S? {
    if (isEmpty())
        return null
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}
return accumulator

* Accumulates value starting with the first element and applying [operation] from left to right
* to current accumulator value and each element.
* Returns `null` if the array is empty.
* @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.
* @sample samples.collections.Collections.Aggregates.reduceOrNull

* @SinceKotlin("1.4")
* @WasExperimental(ExperimentalStdlibApi::class)

public inline fun

ByteArray.reduceOrNull(operation: (acc: Byte, Byte) -> Byte): Byte? {
    if (isEmpty())
        return null

    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator

}

ShortArray.reduceOrNull(operation: (acc: Short, Short) -> Short): Short? {
    if (isEmpty())
        return null

    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator

}

IntArray.reduceOrNull(operation: (acc: Int, Int) -> Int): Int? {
    if (isEmpty())
        return null

    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator

}

LongArray.reduceOrNull(operation: (acc: Long, Long) -> Long): Long? {
    if (isEmpty())
        return null

    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator

}

FloatArray.reduceOrNull(operation: (acc: Float, Float) -> Float): Float? {
    if (isEmpty())
        return null

    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator

}

DoubleArray.reduceOrNull(operation: (acc: Double, Double) -> Double): Double? {
    if (isEmpty())
        return null

    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator

}
BooleanArray.reduceOrNull(operation: (acc: Boolean, Boolean) -> Boolean): Boolean? {
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}

```kotlin
/**
 * Accumulates value starting with the first element and applying [operation] from left to right
 * to current accumulator value and each element.
 * Returns `null` if the array is empty.
 * 
 * @param [operation] function that takes current accumulator value and an element,
 * and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduceOrNull
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public inline fun CharArray.reduceOrNull(operation: (acc: Char, Char) -> Char): Char? {
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the last element and applying [operation] from right to left
 * to each element and current accumulator value.
 * Returns `null` if the array is empty. If the array can be empty in an expected way,
 * please use [reduceRightOrNull] instead. It returns `null` when its receiver is empty.
 * 
 * @param [operation] function that takes an element and current accumulator value,
 * and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduceRight
 */
public inline fun <S, T : S> Array<out T>.reduceRight(operation: (T, acc: S) -> S): S {
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with the last element and applying [operation] from right to left
 * to each element and current accumulator value.
 * Returns `null` if the array is empty. If the array can be empty in an expected way,
 * please use [reduceRightOrNull] instead. It returns `null` when its receiver is empty.
 * 
 * @param [operation] function that takes an element and current accumulator value,
 * and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduceRight
 */
public inline fun ByteArray.reduceRight(operation: (Byte, acc: Byte) -> Byte): Byte {
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with the last element and applying [operation] from right to left
 * to each element and current accumulator value.
 * Returns `null` if the array is empty. If the array can be empty in an expected way,
 * please use [reduceRightOrNull] instead. It returns `null` when its receiver is empty.
 * 
 * @param [operation] function that takes an element and current accumulator value,
 * and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduceRight
 */
public inline fun ShortArray.reduceRight(operation: (Short, acc: Short) -> Short): Short {
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with the last element and applying [operation] from right to left
 * to each element and current accumulator value.
 * Returns `null` if the array is empty. If the array can be empty in an expected way,
 * please use [reduceRightOrNull] instead. It returns `null` when its receiver is empty.
 * 
 * @param [operation] function that takes an element and current accumulator value,
 * and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduceRight
 */
public inline fun IntArray.reduceRight(operation: (Int, acc: Int) -> Int): Int {
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}
```
fun LongArray.reduceRight(operation: (Long, acc: Long) -> Long): Long {
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

fun FloatArray.reduceRight(operation: (Float, acc: Float) -> Float): Float {
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

fun DoubleArray.reduceRight(operation: (Double, acc: Double) -> Double): Double {
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

fun BooleanArray.reduceRight(operation: (Boolean, acc: Boolean) -> Boolean): Boolean {
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

fun CharArray.reduceRight(operation: (Char, acc: Char) -> Char): Char {
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}
/**
 * Accumulates value starting with the last element and applying [operation] from right to
 * left
 * to each element with its index in the original array and current accumulator value.
 * Throws an exception if this array is empty. If the array can be empty in an expected way,
 * please use [reduceRightIndexedOrNull] instead. It returns `null` when its receiver is empty.
 * @param [operation] function that takes the index of an element, the element itself and current accumulator value,
 * and calculates the next accumulator value.
 */
public inline fun ByteArray.reduceRightIndexed(operation: (index: Int, Byte, acc: Byte) -> Byte): Byte {
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.

    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

/**
 * Accumulates value starting with the last element and applying [operation] from right to left
 * to each element with its index in the original array and current accumulator value.
 * Throws an exception if this array is empty. If the array can be empty in an expected way,
 * please use [reduceRightIndexedOrNull] instead. It returns `null` when its receiver is empty.
 * @param [operation] function that takes the index of an element, the element itself and current accumulator value,
 * and calculates the next accumulator value.
 */
public inline fun ShortArray.reduceRightIndexed(operation: (index: Int, Short, acc: Short) -> Short): Short {
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.

    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

/**
 * Accumulates value starting with the last element and applying [operation] from right to left
 * to each element with its index in the original array and current accumulator value.
 * Throws an exception if this array is empty. If the array can be empty in an expected way,
 * please use [reduceRightIndexedOrNull] instead. It returns `null` when its receiver is empty.
 * @param [operation] function that takes the index of an element, the element itself and current accumulator value,
 * and calculates the next accumulator value.
 */
public inline fun IntArray.reduceRightIndexed(operation: (index: Int, Int, acc: Int) -> Int): Int {
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.

    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

/**
 * Accumulates value starting with the last element and applying [operation] from right to left
 * to each element with its index in the original array and current accumulator value.
 * Throws an exception if this array is empty. If the array can be empty in an expected way,
 * please use [reduceRightIndexedOrNull] instead. It returns `null` when its receiver is empty.
 * @param [operation] function that takes the index of an element, the element itself and current accumulator value,
 * and calculates the next accumulator value.
 */
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.

    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

/**
 * Accumulates value starting with the last element and applying [operation] from right to left
 * to each element with its index in the original array and current accumulator value.
 * Throws an exception if this array is empty. If the array can be empty in an expected way,
 * please use [reduceRightIndexedOrNull] instead. It returns `null` when its receiver is empty.
 * @param [operation] function that takes the index of an element, the element itself and current accumulator value,
 * and calculates the next accumulator value.
 */
public inline fun FloatArray.reduceRightIndexed(operation: (index: Int, Float, acc: Float) -> Float): Float {
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.

    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}
Accumulates value starting with the last element and applying \([operation]\) from right to left from each element with its index in the original array and current accumulator value.\n\n**Throws an exception if this array is empty. If the array can be empty in an expected way, please use \([reduceRightIndexedOrNonNull]\) instead. It returns `null` when its receiver is empty.**

**@param \([operation]\) function that takes the index of an element, the element itself and current accumulator value, and calculates the next accumulator value.\n\n**

```kotlin
public inline fun DoubleArray.reduceRightIndexed(operation: (index: Int, Double, acc: Double) -> Double): Double {
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}
```

```kotlin
public inline fun BooleanArray.reduceRightIndexed(operation: (index: Int, Boolean, acc: Boolean) -> Boolean): Boolean {
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}
```

```kotlin
public inline fun CharArray.reduceRightIndexed(operation: (index: Int, Char, acc: Char) -> Char): Char {
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}
```

```kotlin
@SinceKotlin("1.4")
public inline fun <S, T : S> Array<out T>.reduceRightIndexedOrNull(operation: (index: Int, T, acc: S) -> S): S? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator: S = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}
```

```kotlin
@SinceKotlin("1.4")
public inline fun ByteArray.reduceRightIndexedOrNull(operation: (index: Int, Byte, acc: Byte) -> Byte): Byte? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}
```
@param [operation] function that takes the index of an element, the element itself and current accumulator value.
* and calculates the next accumulator value.

and calculates the next accumulator value.

@param [operation] function that takes the index of an element, the element itself and current accumulator value.

and calculates the next accumulator value.

Accumulates value starting with the last element and applying [operation] from right to left to each element with its index in the original array and current accumulator value.

* ln * Returns ‘null’ if the array is empty.

@param [operation] function that takes the index of an element, the element itself and current accumulator value.

* and calculates the next accumulator value.

Accumulates value starting with the last element and applying [operation] from right to left to each element with its index in the original array and current accumulator value.

* Returns ‘null’ if the array is empty.

@param [operation] function that takes the index of an element, the element itself and current accumulator value.

* and calculates the next accumulator value.

Accumulates value starting with the last element and applying [operation] from right to left to each element with its index in the original array and current accumulator value.

* Returns ‘null’ if the array is empty.

@param [operation] function that takes the index of an element, the element itself and current accumulator value.

* and calculates the next accumulator value.

Accumulates value starting with the last element and applying [operation] from right to left to each element with its index in the original array and current accumulator value.

* Returns ‘null’ if the array is empty.

@param [operation] function that takes the index of an element, the element itself and current accumulator value.

* and calculates the next accumulator value.

Accumulates value starting with the last element and applying [operation] from right to left to each element with its index in the original array and current accumulator value.

* Returns ‘null’ if the array is empty.

@param [operation] function that takes the index of an element, the element itself and current accumulator value.

* and calculates the next accumulator value.

Accumulates value starting with the last element and applying [operation] from right to left to each element with its index in the original array and current accumulator value.

* Returns ‘null’ if the array is empty.
@param [operation] function that takes the index of an element, the element itself and current accumulator value.
* and calculates the next accumulator value.

samples.collections.Collections.Aggregates.reduceRightOrNull

CharArray.reduceRightIndexedOrNull(operation: (index: Int, Char, acc: Char) -> Char): Char? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

Accumulates value starting with the last element and applying [operation] from right to left
* to each element and current accumulator value.
* Returns `null` if the array is empty.

samples.collections.Collections.Aggregates.reduceRightOrNull

Array<out T>.reduceRightOrNull(operation: (T, acc: S) -> S): S? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator: S = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

Accumulates value starting with the last element and applying [operation] from right to left
* to each element and current accumulator value.
* Returns `null` if the array is empty.

samples.collections.Collections.Aggregates.reduceRightOrNull

ByteArray.reduceRightOrNull(operation: (Byte, acc: Byte) -> Byte): Byte? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

Accumulates value starting with the last element and applying [operation] from right to left
* to each element and current accumulator value.
* Returns `null` if the array is empty.

samples.collections.Collections.Aggregates.reduceRightOrNull

ShortArray.reduceRightOrNull(operation: (Short, acc: Short) -> Short): Short? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

Accumulates value starting with the last element and applying [operation] from right to left
* to each element and current accumulator value.
* Returns `null` if the array is empty.

samples.collections.Collections.Aggregates.reduceRightOrNull

IntArray.reduceRightOrNull(operation: (Int, acc: Int) -> Int): Int? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

Accumulates value starting with the last element and applying [operation] from right to left
* to each element and current accumulator value.
* Returns `null` if the array is empty.

samples.collections.Collections.Aggregates.reduceRightOrNull

LongArray.reduceRightOrNull(operation: (Long, acc: Long) -> Long): Long? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

Accumulates value starting with the last element and applying [operation] from right to left
* to each element and current accumulator value.
* Returns `null` if the array is empty.

samples.collections.Collections.Aggregates.reduceRightOrNull

accumulator value,\n * and calculates the next accumulator value.\n
```kotlin
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public inline fun FloatArray.reduceRightOrNull(operation: (Float, acc: Float) -> Float): Float? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(accumulator, get(index--))
    }
    return accumulator
}
```

```kotlin
/**
 * Accumulates value starting with the last element and applying [operation] from right to left
 * to each element and current accumulator value.\n
 * @param [operation] function that takes an element and current accumulator value,\n * and calculates the next accumulator value.\n
 * @sample samples.collections.Collections.Aggregates.reduceRightOrNull
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public inline fun DoubleArray.reduceRightOrNull(operation: (Double, acc: Double) -> Double): Double? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(accumulator, get(index--))
    }
    return accumulator
}
```

```kotlin
/**
 * Accumulates value starting with the last element and applying [operation] from right to left
 * to each element and current accumulator value.\n
 * @param [operation] function that takes an element and current accumulator value,\n * and calculates the next accumulator value.\n
 * @sample samples.collections.Collections.Aggregates.reduceRightOrNull
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public inline fun BooleanArray.reduceRightOrNull(operation: (Boolean, acc: Boolean) -> Boolean): Boolean? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(accumulator, get(index--))
    }
    return accumulator
}
```

```kotlin
/**
 * Accumulates value starting with the last element and applying [operation] from right to left
 * to each element and current accumulator value.\n
 * @param [operation] function that takes an element and current accumulator value,\n * and calculates the next accumulator value.\n
 * @sample samples.collections.Collections.Aggregates.reduceRightOrNull
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public inline fun CharArray.reduceRightOrNull(operation: (Char, acc: Char) -> Char): Char? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(accumulator, get(index--))
    }
    return accumulator
}
```

```kotlin
/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right
 * to each element and current accumulator value that starts with [initial] value.\n
 * Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.\n
 * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n
 * @sample samples.collections.Collections.Aggregates.runningFold
 */
@SinceKotlin("1.4")
public inline fun <T, R> Array<out T>.runningFold(initial: R, operation: (acc: R, T) -> R): List<R> {
    if (isEmpty()) return listOf(initial)
    val result = ArrayList<R>(size + 1).apply { add(initial) }
    var accumulator = initial
    for (element in this) {
        accumulator = operation(accumulator, element)
        result.add(accumulator)
    }
    return result
}
```

```kotlin
/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right
 * to each element and current accumulator value that starts with [initial] value.\n
 * Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.\n
 * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n
 * @sample samples.collections.Collections.Aggregates.runningFold
 */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun <R> ByteArray.runningFold(initial: R, operation: (acc: R, Byte) -> R): List<R> {
    if (isEmpty()) return listOf(initial)
    val result = ArrayList<R>(size + 1).apply { add(initial) }
    var accumulator = initial
    for (element in this) {
        accumulator = operation(accumulator, element)
        result.add(accumulator)
    }
    return result
}
```

```kotlin
/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right
 * to each element and current accumulator value that starts with [initial] value.\n
 * Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.\n
 * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n
 * @sample samples.collections.Collections.Aggregates.runningFold
 */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun <R> IntArray.runningFold(initial: R, operation: (acc: R, Int) -> R): List<R> {
    if (isEmpty()) return listOf(initial)
    val result = ArrayList<R>(size + 1).apply { add(initial) }
    var accumulator = initial
    for (element in this) {
        accumulator = operation(accumulator, element)
        result.add(accumulator)
    }
    return result
}
```

```kotlin
/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right
 * to each element and current accumulator value that starts with [initial] value.\n
 * Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.\n
 * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n
 * @sample samples.collections.Collections.Aggregates.runningFold
 */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
    if (isEmpty()) return listOf(initial)
    val result = ArrayList<R>(size + 1).apply { add(initial) }
    var accumulator = initial
    for (element in this) {
        accumulator = operation(accumulator, element)
        result.add(accumulator)
    }
    return result
}
```

```kotlin
/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right
 * to each element and current accumulator value that starts with [initial] value.\n
 * Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.\n
 * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n
 * @sample samples.collections.Collections.Aggregates.runningFold
 */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun <R> ShortArray.runningFold(initial: R, operation: (acc: R, Short) -> R): List<R> {
    if (isEmpty()) return listOf(initial)
    val result = ArrayList<R>(size + 1).apply { add(initial) }
    var accumulator = initial
    for (element in this) {
        accumulator = operation(accumulator, element)
        result.add(accumulator)
    }
    return result
}
```
containing successive accumulation values generated by applying [operation] from left to right
and current accumulator value that starts with [initial] value.

Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.

@param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.

@sample samples.collections.Collections.Aggregates.runningFold

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right
 * and current accumulator value that starts with [initial] value.
 * 
 * Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.
 * 
 * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.runningFold

/*@SinceKotlin("1.4")*/
@kotlin.internal.InlineOnly
public inline fun <R> ShortArray.runningFold(initial: R, operation: (acc: R, Short) -> R): List<R> {
    if (isEmpty()) return listOf(initial)
    val result = ArrayList<R>(size + 1).apply { add(initial) }
    var accumulator = initial
    for (element in this) {
        accumulator = operation(accumulator, element)
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right
 * and current accumulator value that starts with [initial] value.
 * 
 * Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.
 * 
 * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.runningFold

/*@SinceKotlin("1.4")*/
@kotlin.internal.InlineOnly
public inline fun <R> IntArray.runningFold(initial: R, operation: (acc: R, Int) -> R): List<R> {
    if (isEmpty()) return listOf(initial)
    val result = ArrayList<R>(size + 1).apply { add(initial) }
    var accumulator = initial
    for (element in this) {
        accumulator = operation(accumulator, element)
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right
 * and current accumulator value that starts with [initial] value.
 * 
 * Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.
 * 
 * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.runningFold

/*@SinceKotlin("1.4")*/
@kotlin.internal.InlineOnly
    if (isEmpty()) return listOf(initial)
    val result = ArrayList<R>(size + 1).apply { add(initial) }
    var accumulator = initial
    for (element in this) {
        accumulator = operation(accumulator, element)
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right
 * and current accumulator value that starts with [initial] value.
 * 
 * Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.
 * 
 * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.runningFold

/*@SinceKotlin("1.4")*/
@kotlin.internal.InlineOnly
public inline fun <R> FloatArray.runningFold(initial: R, operation: (acc: R, Float) -> R): List<R> {
    if (isEmpty()) return listOf(initial)
    val result = ArrayList<R>(size + 1).apply { add(initial) }
    var accumulator = initial
    for (element in this) {
        accumulator = operation(accumulator, element)
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right
 * and current accumulator value that starts with [initial] value.
 * 
 * Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.
 * 
 * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.runningFold

/*@SinceKotlin("1.4")*/
@kotlin.internal.InlineOnly
public inline fun <R> DoubleArray.runningFold(initial: R, operation: (acc: R, Double) -> R): List<R> {
    if (isEmpty()) return listOf(initial)
    val result = ArrayList<R>(size + 1).apply { add(initial) }
    var accumulator = initial
    for (element in this) {
        accumulator = operation(accumulator, element)
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right
 * and current accumulator value that starts with [initial] value.
 * 
 * Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.
 * 
 * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.runningFold

/*@SinceKotlin("1.4")*/
@kotlin.internal.InlineOnly
public inline fun <R> Array<@UnsafeVariance R>.runningFold(initial: R, operation: (acc: R, @UnsafeVariance R?) -> @UnsafeVariance R): List<@UnsafeVariance R> {
    if (isEmpty()) return listOf(initial)
    val result = ArrayList<@UnsafeVariance R>(size + 1).apply { add(initial) }
    var accumulator = initial
    for (element in this) {
        accumulator = operation(accumulator, element)
        result.add(accumulator)
    }
    return result
}
@param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.

```kotlin
*@

@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun <R> BooleanArray.runningFold(initial: R, operation: (acc: R, Boolean) -> R): List<R>
```

Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element and current accumulator value that starts with [initial] value.

```kotlin
*@

@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun <R> CharArray.runningFold(initial: R, operation: (acc: R, Char) -> R): List<R>
```

```kotlin
*@

@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
```

```kotlin
*@

@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
```

Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.

```kotlin
*@

@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
```

```kotlin
*@

@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
```

Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.

```kotlin
*@

@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
```

Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.

```kotlin
*@

@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
```

Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.
function that takes the index of an element, current accumulator value \( \text{\texttt{acc}} \) and the element itself, and calculates the next accumulator value.

```kotlin
/*\@SinceKotlin("1.4")*/fun@kotlin.internal.InlineOnly\npublic inline fun <R>
IntArray.runningFoldIndexed(initial: R, operation: (Int, acc: R, Int) -> R): List<R> {
    \n    if (isEmpty()) return val result = List\<R\>(size + 1).apply { add\(\) } var accumulator = initial
    \n    for (index in indices)
        result.add(accumulator)
    \n    return result
}
```

```kotlin
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun <R>
LongArray.runningFoldIndexed(initial: R, operation: (Int, acc: R, Long) -> R): List<R> {
    \n    if (isEmpty()) return val result = List\<R\>(size + 1).apply { add\(\) } var accumulator = initial
    \n    for (index in indices)
        result.add(accumulator)
    \n    return result
}
```

```kotlin
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun <R>
FloatArray.runningFoldIndexed(initial: R, operation: (Int, acc: R, Float) -> R): List<R> {
    \n    if (isEmpty()) return val result = List\<R\>(size + 1).apply { add\(\) } var accumulator = initial
    \n    for (index in indices)
        result.add(accumulator)
    \n    return result
}
```

```kotlin
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun <R>
DoubleArray.runningFoldIndexed(initial: R, operation: (Int, acc: R, Double) -> R): List<R> {
    \n    if (isEmpty()) return val result = List\<R\>(size + 1).apply { add\(\) } var accumulator = initial
    \n    for (index in indices)
        result.add(accumulator)
    \n    return result
}
```

```kotlin
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun <R>
BooleanArray.runningFoldIndexed(initial: R, operation: (Int, acc: R, Boolean) -> R): List<R> {
    \n    if (isEmpty()) return val result = List\<R\>(size + 1).apply { add\(\) } var accumulator = initial
    \n    for (index in indices)
        result.add(accumulator)
    \n    return result
}
```

```kotlin
/*\@SinceKotlin("1.4")*/
```
current accumulator value that starts with [initial] value. Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list. The function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.

```kotlin
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
@public inline fun <R>
CharArray.runningFoldIndexed(initial: R, operation: (index: Int, acc: R, Char) -> R): List<R> {
    if (isEmpty())
        return listOf(initial)
    val result = ArrayList<R>(size + 1).apply {
        add(initial)
    }
    var accumulator = initial
    for (index in indices) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element and current accumulator value that starts with the first element of this array. Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.
 * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.
 * @sample samples.collections.Aggregates.runningReduce
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@public inline fun <S, T : S>
Array<out T>.runningReduce(operation: (acc: S, T) -> S): List<S> {
    if (isEmpty())
        return emptyList()
    var accumulator: S = this[0]
    val result = ArrayList<S>(size).apply {
        add(accumulator)
    }
    for (index in 1 until size) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element and current accumulator value that starts with the first element of this array.
 * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.
 * @sample samples.collections.Aggregates.runningReduce
 */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
@public inline fun ByteArray.runningReduce(operation: (acc: Byte, Byte) -> Byte): List<Byte> {
    if (isEmpty())
        return emptyList()
    var accumulator = this[0]
    val result = ArrayList<Byte>(size).apply {
        add(accumulator)
    }
    for (index in 1 until size) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element and current accumulator value that starts with the first element of this array.
 * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.
 * @sample samples.collections.Aggregates.runningReduce
 */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
@public inline fun ShortArray.runningReduce(operation: (acc: Short, Short) -> Short): List<Short> {
    if (isEmpty())
        return emptyList()
    var accumulator = this[0]
    val result = ArrayList<Short>(size).apply {
        add(accumulator)
    }
    for (index in 1 until size) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element and current accumulator value that starts with the first element of this array.
 * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.
 * @sample samples.collections.Aggregates.runningReduce
 */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
@public inline fun IntArray.runningReduce(operation: (acc: Int, Int) -> Int): List<Int> {
    if (isEmpty())
        return emptyList()
    var accumulator = this[0]
    val result = ArrayList<Int>(size).apply {
        add(accumulator)
    }
    for (index in 1 until size) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element and current accumulator value that starts with the first element of this array.
 * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.
 * @sample samples.collections.Aggregates.runningReduce
 */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
@public inline fun LongArray.runningReduce(operation: (acc: Long, Long) -> Long): List<Long> {
    if (isEmpty())
        return emptyList()
    var accumulator = this[0]
    val result = ArrayList<Long>(size).apply {
        add(accumulator)
    }
    for (index in 1 until size) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}
```
val result = ArrayList<Long>(size).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right.
 * Takes current accumulator value and an element, and calculates the next accumulator value.
 */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun FloatArray.runningReduce(operation: (acc: Float, Float) -> Float): List<Float> {
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<Float>(size).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right.
 * Takes current accumulator value and an element, and calculates the next accumulator value.
 */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun DoubleArray.runningReduce(operation: (acc: Double, Double) -> Double): List<Double> {
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<Double>(size).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right.
 * Takes current accumulator value and an element, and calculates the next accumulator value.
 */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun BooleanArray.runningReduce(operation: (acc: Boolean, Boolean) -> Boolean): List<Boolean> {
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<Boolean>(size).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right.
 * Takes current accumulator value and an element, and calculates the next accumulator value.
 */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun CharArray.runningReduce(operation: (acc: Char, Char) -> Char): List<Char> {
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<Char>(size).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right.
 * Takes current accumulator value and an element, and calculates the next accumulator value.
 */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun <S, T : S> Array<out T>.runningReduceIndexed(operation: (index: Int, acc: S, T) -> S): List<S> {
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<S>(size).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun ByteArray.runningReduceIndexed(operation: (index: Int, acc: Byte, Byte) -> Byte): List<Byte> {
    if (isEmpty())
        return emptyList()
    var accumulator = this[0]
    val result = ArrayList<Byte>(size).apply {
        add(accumulator)
    }
    for (index in 1 until size) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right
 * to each element, its index in the original array and current accumulator value that starts with the first element of this array.
 * @param [operation] function that takes the index of an element, current accumulator value
 * and the element itself, and calculates the next accumulator value.
 * @sample samples.collections.Aggregates.runningReduce
 */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun ShortArray.runningReduceIndexed(operation: (index: Int, acc: Short, Short) -> Short): List<Short> {
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<Short>(size).apply {
        add(accumulator)
    }
    for (index in 1 until size) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right
 * to each element, its index in the original array and current accumulator value that starts with the first element of this array.
 * @param [operation] function that takes the index of an element, current accumulator value
 * and the element itself, and calculates the next accumulator value.
 * @sample samples.collections.Aggregates.runningReduce
 */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun IntArray.runningReduceIndexed(operation: (index: Int, acc: Int, Int) -> Int): List<Int> {
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<Int>(size).apply {
        add(accumulator)
    }
    for (index in 1 until size) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right
 * to each element, its index in the original array and current accumulator value that starts with the first element of this array.
 * @param [operation] function that takes the index of an element, current accumulator value
 * and the element itself, and calculates the next accumulator value.
 * @sample samples.collections.Aggregates.runningReduce
 */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<Long>(size).apply {
        add(accumulator)
    }
    for (index in 1 until size) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right
 * to each element, its index in the original array and current accumulator value that starts with the first element of this array.
 * @param [operation] function that takes the index of an element, current accumulator value
 * and the element itself, and calculates the next accumulator value.
 * @sample samples.collections.Aggregates.runningReduce
 */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun FloatArray.runningReduceIndexed(operation: (index: Int, acc: Float, Float) -> Float): List<Float> {
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<Float>(size).apply {
        add(accumulator)
    }
    for (index in 1 until size) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right
 * to each element, its index in the original array and current accumulator value that starts with the first element of this array.
 * @param [operation] function that takes the index of an element, current accumulator value
 * and the element itself, and calculates the next accumulator value.
 * @sample samples.collections.Aggregates.runningReduce
 */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun DoubleArray.runningReduceIndexed(operation: (index: Int, acc: Double, Double) -> Double): List<Double> {
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<Double>(size).apply {
        add(accumulator)
    }
    for (index in 1 until size) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

Open Source Used In security_automation identity-guest-cs 1314
add(accumulator) for (index in 1 until size) { accumulator = operation(index, accumulator, this[index]) result.add(accumulator) return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element, its index in the original array and current accumulator value that starts with the first element of this array.
 * @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.
 * @param samples.collections.Collections.Aggregates.runningReduce
 */
@SinceKotlin("1.4") fun @kotlin.internal.InlineOnly fun

BooleanArray.runningReduceIndexed(operation: (index: Int, acc: Boolean, Boolean) -> Boolean): List<Boolean> {
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<Boolean>(size).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element, its index in the original array and current accumulator value that starts with the first element of this array.
 * @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.
 * @param samples.collections.Collections.Aggregates.runningReduce
 */
@SinceKotlin("1.4") fun @kotlin.internal.InlineOnly fun

CharArray.runningReduceIndexed(operation: (index: Int, acc: Char, Char) -> Char): List<Char> {
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<Char>(size).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element and current accumulator value that starts with [initial] value.
 * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.
 * @param samples.collections.Collections.Aggregates.scan
 */
@SinceKotlin("1.4") fun @WasExperimental(ExperimentalStdlibApi::class) fun <T, R> Array<out T>.scan(initial: R, operation: (acc: R, T) -> R): List<R> {
    return runningFold(initial, operation)
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element and current accumulator value that starts with [initial] value.
 * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.
 * @param samples.collections.Collections.Aggregates.scan
 */
@SinceKotlin("1.4") fun @WasExperimental(ExperimentalStdlibApi::class) fun <R> ByteArray.scan(initial: R, operation: (acc: R, Byte) -> R): List<R> {
    return runningFold(initial, operation)
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element and current accumulator value that starts with [initial] value.
 * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.
 * @param samples.collections.Collections.Aggregates.scan
 */
@SinceKotlin("1.4") fun @WasExperimental(ExperimentalStdlibApi::class) fun <R> ShortArray.scan(initial: R, operation: (acc: R, Short) -> R): List<R> {
    return runningFold(initial, operation)
}
/**
 * Returns a list containing successive accumulation values generated by applying [operation]
 * from left to right to each element and current accumulator value that starts with [initial] value.
 * Note that ‘acc’ value passed to [operation] function should not be mutated; otherwise it would affect
 * the previous value in resulting list.
 * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.scan
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@kotlin.internal.InlineOnly
public inline fun <T, R> CharArray.scan(initial: R, operation: (acc: R, Char) -> R): List<R> {
    return runningFold(initial, operation)
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation]
 * from left to right to each element and current accumulator value that starts with [initial] value.
 * Note that ‘acc’ value passed to [operation] function should not be mutated; otherwise it would affect
 * the previous value in resulting list.
 * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.scan
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@kotlin.internal.InlineOnly
public inline fun <R> CharArray.scan(initial: R, operation: (acc: R, Char) -> R): List<R> {
    return runningFold(initial, operation)
}
Array<out T>.scanIndexed(initial: R, operation: (index: Int, acc: R, T) -> R): List<R> {\n    return runningFoldIndexed(initial, operation)\n}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element, its index in the original array and current accumulator value that starts with [initial] value.\n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n */
@SinceKotlin("1.4")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic inline fun <R> ByteArray.scanIndexed(initial: R, operation: (index: Int, acc: R, Byte) -> R): List<R> {\n    return runningFoldIndexed(initial, operation)\n}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element, its index in the original array and current accumulator value that starts with [initial] value.\n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n */
@SinceKotlin("1.4")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic inline fun <R> ShortArray.scanIndexed(initial: R, operation: (index: Int, acc: R, Short) -> R): List<R> {\n    return runningFoldIndexed(initial, operation)\n}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element, its index in the original array and current accumulator value that starts with [initial] value.\n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n */
@SinceKotlin("1.4")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic inline fun <R> IntArray.scanIndexed(initial: R, operation: (index: Int, acc: R, Int) -> R): List<R> {\n    return runningFoldIndexed(initial, operation)\n}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element, its index in the original array and current accumulator value that starts with [initial] value.\n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n */
@SinceKotlin("1.4")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic inline fun <R> LongArray.scanIndexed(initial: R, operation: (index: Int, acc: R, Long) -> R): List<R> {\n    return runningFoldIndexed(initial, operation)\n}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element, its index in the original array and current accumulator value that starts with [initial] value.\n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n */
@SinceKotlin("1.4")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic inline fun <R> FloatArray.scanIndexed(initial: R, operation: (index: Int, acc: R, Float) -> R): List<R> {\n    return runningFoldIndexed(initial, operation)\n}

/*
* @SinceKotlin("1.4")\n* @WasExperimental(ExperimentalStdlibApi::class)\n* 
* public inline fun <R> CharArray.scanIndexed(initial: R, operation: (index: Int, acc: R, Char) -> R): List<R> {\n*    return runningFoldIndexed(initial, operation)\n*}
*/

*/
@SinceKotlin("1.4")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic inline fun <R> CharArray.scanIndexed(initial: R, operation: (index: Int, acc: R, Char) -> R): List<R> {\n    return runningFoldIndexed(initial, operation)\n}
inline fun <R> DoubleArray.scanIndexed(initial: R, operation: (index: Int, acc: R, Double) -> R): List<R> {
    return runningFoldIndexed(initial, operation)\n        * Returns a list containing successive accumulation values
        * generated by applying [operation] from left to right\n        * to each element, its index in the original array and current
        * accumulator value that starts with [initial] value.\n        * Note that `acc` value passed to [operation] function should
        * otherwise it would affect the previous value in resulting list.\n        * * @param [operation] function that takes the index of an element,
c inline fun BooleanArray.scanReduce(operation: (acc: Boolean, Boolean) -> Boolean): List<Boolean> {
    return runningReduce(operation)
}

@Deprecated("Use runningReduce instead.",
    ReplaceWith("runningReduce(operation)"), level =
    DeprecationLevel.ERROR)
@ExperimentalStdlibApi
public inline fun <S, T : S>
    Array<out T>.scanReduceIndexed(operation: (index: Int, acc: S, T) -> S): List<S> {
    return runningReduceIndexed(operation)
}

@Deprecated("Use runningReduceIndexed instead.",
    ReplaceWith("runningReduceIndexed(operation)"), level =
    DeprecationLevel.ERROR)
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
public inline fun CharArray.scanReduceIndexed(operation: (index: Int, acc: Char, Char) -> Char): List<Char> {
    return runningReduceIndexed(operation)
}

/**
 * Returns the sum of all values produced by [selector] function
 * applied to each element in the array.
 */
public inline fun <T> Array<out T>.sumBy(selector: (T) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun ByteArray.sumBy(selector: (Byte) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector]
 * function applied to each element in the array.
 */
public inline fun ShortArray.sumBy(selector: (Short) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun IntArray.sumBy(selector: (Int) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector]
 * function applied to each element in the array.
 */
public inline fun LongArray.sumBy(selector: (Long) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector]
 * function applied to each element in the array.
 */
public inline fun FloatArray.sumBy(selector: (Float) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector]
 * function applied to each element in the array.
 */
public inline fun DoubleArray.sumBy(selector: (Double) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun BooleanArray.scanReduceIndexed(operation: (index: Int, acc: Boolean, Boolean) -> Boolean): List<Boolean> {
    return runningReduceIndexed(operation)
}

/**
 * Returns the sum of all values produced by [selector]
 * function applied to each element in the array.
 */
public inline fun CharArray.scanReduceIndexed(operation: (index: Int, acc: Char, Char) -> Char): List<Char> {
    return runningReduceIndexed(operation)
}
```kotlin
{\n    var sum: Int = 0\n    for (element in this) {\n        sum += selector(element)\n    }\n    return sum\n}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.\n */
public inline fun IntArray.sumBy(selector: (Int) -> Int): Int {
    var sum: Int = 0\n    for (element in this) {
        sum += selector(element)\n    }\n    return sum\n}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.\n */
public inline fun LongArray.sumBy(selector: (Long) -> Int): Int {
    var sum: Int = 0\n    for (element in this) {
        sum += selector(element)\n    }\n    return sum\n}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.\n */
public inline fun FloatArray.sumBy(selector: (Float) -> Int): Int {
    var sum: Int = 0\n    for (element in this) {
        sum += selector(element)\n    }\n    return sum\n}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.\n */
public inline fun DoubleArray.sumBy(selector: (Double) -> Int): Int {
    var sum: Int = 0\n    for (element in this) {
        sum += selector(element)\n    }\n    return sum\n}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.\n */
public inline fun BooleanArray.sumBy(selector: (Boolean) -> Int): Int {
    var sum: Int = 0\n    for (element in this) {
        sum += selector(element)\n    }\n    return sum\n}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.\n */
public inline fun CharArray.sumBy(selector: (Char) -> Int): Int {
    var sum: Int = 0\n    for (element in this) {
        sum += selector(element)\n    }\n    return sum\n}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.\n */
public inline fun <T> Array<out T>.sumByDouble(selector: (T) -> Double): Double {
    var sum: Double = 0.0\n    for (element in this) {
        sum += selector(element)\n    }\n    return sum\n}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.\n */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@kotlin.jvm.JvmName("sumOfDouble")
@kotlin.internal.InlineOnly
public inline fun <T> Array<out T>.sumOf(selector: (T) -> Double): Double {
    var sum: Double = 0.toDouble()\n    for (element in this) {
        sum += selector(element)\n    }\n    return sum\n}
```

Open Source Used in security_automation identity-guest-cs 1320
ByLambdaReturnType
@kotlin.jvm.JvmName("sumOfDouble")
@kotlin.internal.InlineOnly
public inline fun ByteArray.sumOf(selector: (Byte) -> Double): Double {
    var sum: Double = 0.toDouble()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by 
 * [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("sumOfDouble")
@kotlin.internal.InlineOnly
public inline fun ShortArray.sumOf(selector: (Short) -> Double): Double {
    var sum: Double = 0.toDouble()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by 
 * [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("sumOfDouble")
@kotlin.internal.InlineOnly
public inline fun IntArray.sumOf(selector: (Int) -> Double): Double {
    var sum: Double = 0.toDouble()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by 
 * [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("sumOfDouble")
@kotlin.internal.InlineOnly
public inline fun LongArray.sumOf(selector: (Long) -> Double): Double {
    var sum: Double = 0.toDouble()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by 
 * [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("sumOfDouble")
@kotlin.internal.InlineOnly
public inline fun FloatArray.sumOf(selector: (Float) -> Double): Double {
    var sum: Double = 0.toDouble()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by 
 * [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("sumOfDouble")
@kotlin.internal.InlineOnly
public inline fun DoubleArray.sumOf(selector: (Double) -> Double): Double {
    var sum: Double = 0.toDouble()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by 
 * [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("sumOfDouble")
@kotlin.internal.InlineOnly
public inline fun BooleanArray.sumOf(selector: (Boolean) -> Double): Double {
    var sum: Double = 0.toDouble()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by 
 * [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("sumOfDouble")
@kotlin.internal.InlineOnly
public inline fun CharArray.sumOf(selector: (Char) -> Double): Double {
    var sum: Double = 0.toDouble()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by 
 * [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("sumOfDouble")
@kotlin.internal.InlineOnly
public inline fun <T> Array<T>.sumOf(selector: (T) -> Int): Int {
    var sum: Int = 0.toInt()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by 
 * [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("sumOfDouble")
@kotlin.internal.InlineOnly
public inline fun ByteArray.sumOf(selector: (Byte) -> Int): Int {
    var sum: Int = 0.toInt()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}
applied to each element in the array.

```kotlin
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("sumOfInt")
@kotlin.internal.InlineOnly
public inline fun ShortArray.sumOf(selector: (Short) -> Int): Int {
    var sum: Int = 0.toInt()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("sumOfInt")
@kotlin.internal.InlineOnly
public inline fun IntArray.sumOf(selector: (Int) -> Int): Int {
    var sum: Int = 0.toInt()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("sumOfInt")
@kotlin.internal.InlineOnly
public inline fun LongArray.sumOf(selector: (Long) -> Int): Int {
    var sum: Int = 0.toInt()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("sumOfInt")
@kotlin.internal.InlineOnly
public inline fun FloatArray.sumOf(selector: (Float) -> Int): Int {
    var sum: Int = 0.toInt()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("sumOfInt")
@kotlin.internal.InlineOnly
public inline fun DoubleArray.sumOf(selector: (Double) -> Int): Int {
    var sum: Int = 0.toInt()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("sumOfLong")
@kotlin.internal.InlineOnly
public inline fun ByteArray.sumOf(selector: (Byte) -> Long): Long {
    var sum: Long = 0.toLong()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("sumOfLong")
@kotlin.internal.InlineOnly
public inline fun <T> Array<T>.sumOf(selector: (T) -> Long): Long {
    var sum: Long = 0.toLong()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("sumOfLong")
@kotlin.internal.InlineOnly
public inline fun BooleanArray.sumOf(selector: (Boolean) -> Int): Int {
    var sum: Int = 0.toInt()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("sumOfLong")
@kotlin.internal.InlineOnly
public inline fun CharArray.sumOf(selector: (Char) -> Int): Int {
    var sum: Int = 0.toInt()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("sumOfLong")
@kotlin.internal.InlineOnly
public inline fun <T> Array<T>.sumOf(selector: (T) -> Long): Long {
    var sum: Long = 0.toLong()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("sumOfLong")
@kotlin.internal.InlineOnly
public inline fun ByteArray.sumOf(selector: (Byte) -> Long): Long {
    var sum: Long = 0.toLong()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("sumOfLong")
@kotlin.internal.InlineOnly
public inline fun
```
ShortArray.sumOf(selector: (Short) -> Long): Long

```kotlin
var sum: Long = 0.toLong()
for (element in this) {
    sum += selector(element)
}
return sum
```

* Returns the sum of all values produced by [selector] function applied to each element in the array.

* Returns the sum of all values produced by [selector] function applied to each element in the array.

### Array Sum Methods

- **IntArray**: 
  ```kotlin
  IntArray.sumOf(selector: (Int) -> Long): Long
  ```
  * Returns the sum of all values produced by [selector] function applied to each element in the array.

- **LongArray**: 
  ```kotlin
  LongArray.sumOf(selector: (Long) -> Long): Long
  ```
  * Returns the sum of all values produced by [selector] function applied to each element in the array.

- **FloatArray**: 
  ```kotlin
  FloatArray.sumOf(selector: (Float) -> Long): Long
  ```
  * Returns the sum of all values produced by [selector] function applied to each element in the array.

- **DoubleArray**: 
  ```kotlin
  DoubleArray.sumOf(selector: (Double) -> Long): Long
  ```
  * Returns the sum of all values produced by [selector] function applied to each element in the array.

- **BooleanArray**: 
  ```kotlin
  BooleanArray.sumOf(selector: (Boolean) -> Long): Long
  ```
  * Returns the sum of all values produced by [selector] function applied to each element in the array.

- **CharArray**: 
  ```kotlin
  CharArray.sumOf(selector: (Char) -> Long): Long
  ```
  * Returns the sum of all values produced by [selector] function applied to each element in the array.

- **ByteArray**: 
  ```kotlin
  ByteArray.sumOf(selector: (Byte) -> UInt): UInt
  ```
  * Returns the sum of all values produced by [selector] function applied to each element in the array.

- **ShortArray**: 
  ```kotlin
  ShortArray.sumOf(selector: (Short) -> UInt): UInt
  ```
  * Returns the sum of all values produced by [selector] function applied to each element in the array.

- **TArray**: 
  ```kotlin
  TArray.sumOf(selector: (T) -> UInt): UInt
  ```
  * Returns the sum of all values produced by [selector] function applied to each element in the array.

- **TArray**: 
  ```kotlin
  TArray.sumOf(selector: (T) -> UInt): UInt
  ```
  * Returns the sum of all values produced by [selector] function applied to each element in the array.
public inline fun IntArray.sumOf(selector: (Int) -> UInt): UInt {
    var sum: UInt = 0.toUInt()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun LongArray.sumOf(selector: (Long) -> UInt): UInt {
    var sum: UInt = 0.toUInt()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun FloatArray.sumOf(selector: (Float) -> UInt): UInt {
    var sum: UInt = 0.toUInt()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun DoubleArray.sumOf(selector: (Double) -> UInt): UInt {
    var sum: UInt = 0.toUInt()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun BooleanArray.sumOf(selector: (Boolean) -> UInt): UInt {
    var sum: UInt = 0.toUInt()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun CharArray.sumOf(selector: (Char) -> UInt): UInt {
    var sum: UInt = 0.toUInt()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun <T> Array<out T>.sumOf(selector: (T) -> ULong): ULong {
    var sum: ULong = 0.toULong()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun ByteArray.sumOf(selector: (Byte) -> ULong): ULong {
    var sum: ULong = 0.toULong()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun ShortArray.sumOf(selector: (Short) -> ULong): ULong {
    var sum: ULong = 0.toULong()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun IntArray.sumOf(selector: (Int) -> ULong): ULong {
    var sum: ULong = 0.toULong()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun LongArray.sumOf(selector: (Long) -> ULong): ULong {
    var sum: ULong = 0.toULong()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun FloatArray.sumOf(selector: (Float) -> ULong): ULong {
    var sum: ULong = 0.toULong()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun DoubleArray.sumOf(selector: (Double) -> ULong): ULong {
    var sum: ULong = 0.toULong()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun BooleanArray.sumOf(selector: (Boolean) -> ULong): ULong {
    var sum: ULong = 0.toULong()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun CharArray.sumOf(selector: (Char) -> ULong): ULong {
    var sum: ULong = 0.toULong()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun <T> Array<out T>.sumOf(selector: (T) -> ULong): ULong {
    var sum: ULong = 0.toULong()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun ByteArray.sumOf(selector: (Byte) -> ULong): ULong {
    var sum: ULong = 0.toULong()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun ShortArray.sumOf(selector: (Short) -> ULong): ULong {
    var sum: ULong = 0.toULong()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}
0.toULong() for (element in this) { \n    sum += selector(element) \n  } return sum\n} \n/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")@OptIn(kotlin.experimental.ExperimentalTypeInference::class)@OverloadResolution
ByLambdaReturnType@kotlin.jvm.JvmName("sumOfULong")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly
public inline fun LongArray.sumOf(selector: (Long) -> ULong): ULong { \n    var sum: ULong = 0.toULong() \n    for (element in this) { \n        sum += selector(element) \n    } return sum\n} \n/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")@OptIn(kotlin.experimental.ExperimentalTypeInference::class)@OverloadResolution
ByLambdaReturnType@kotlin.jvm.JvmName("sumOfULong")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly
public inline fun FloatArray.sumOf(selector: (Float) -> ULong): ULong { \n    var sum: ULong = 0.toULong() \n    for (element in this) { \n        sum += selector(element) \n    } return sum\n} \n/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")@OptIn(kotlin.experimental.ExperimentalTypeInference::class)@OverloadResolution
ByLambdaReturnType@kotlin.jvm.JvmName("sumOfULong")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly
public inline fun DoubleArray.sumOf(selector: (Double) -> ULong): ULong { \n    var sum: ULong = 0.toULong() \n    for (element in this) { \n        sum += selector(element) \n    } return sum\n} \n/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")@OptIn(kotlin.experimental.ExperimentalTypeInference::class)@OverloadResolution
ByLambdaReturnType@kotlin.jvm.JvmName("sumOfULong")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly
public inline fun BooleanArray.sumOf(selector: (Boolean) -> ULong): ULong { \n    var sum: ULong = 0.toULong() \n    for (element in this) { \n        sum += selector(element) \n    } return sum\n} \n/**
 * Returns an original collection containing all the non-`null` elements, throwing an [IllegalArgumentExce}
**
 * Splits the original array into pair of lists,\n * where *first* list contains elements for which 
 [predicate] yielded `true`,\n * while *second* list contains elements for which [predicate] yielded `false`.\n * \n @sample samples.collections.Arrays.Transformations.partitionArrayOfPrimitives
 */
public inline fun IntArray.partition(predicate: (Int) -> Boolean): Pair<List<Int>, List<Int>> {
    val first = ArrayList<Int>()
    val second = ArrayList<Int>()
    for (element in this) {
        if (predicate(element)) {
            first.add(element)
        } else {
            second.add(element)
        }
    }
    return Pair(first, second)
}

/**
 * Splits the original array into pair of lists,\n * where *first* list contains elements for which [predicate] yielded `true`,\n * while *second* list contains elements for which [predicate] yielded `false`.\n * \n @sample samples.collections.Arrays.Transformations.partitionArrayOfPrimitives
 */
public inline fun LongArray.partition(predicate: (Long) -> Boolean): Pair<List<Long>, List<Long>> {
    val first = ArrayList<Long>()
    val second = ArrayList<Long>()
    for (element in this) {
        if (predicate(element)) {
            first.add(element)
        } else {
            second.add(element)
        }
    }
    return Pair(first, second)
}

/**
 * Splits the original array into pair of lists,\n * where *first* list contains elements for which 
 [predicate] yielded `true`,\n * while *second* list contains elements for which [predicate] yielded `false`.\n * \n @sample samples.collections.Arrays.Transformations.partitionArrayOfPrimitives
 */
public inline fun FloatArray.partition(predicate: (Float) -> Boolean): Pair<List<Float>, List<Float>> {
    val first = ArrayList<Float>()
    val second = ArrayList<Float>()
    for (element in this) {
        if (predicate(element)) {
            first.add(element)
        } else {
            second.add(element)
        }
    }
    return Pair(first, second)
}

/**
 * Splits the original array into pair of lists,\n * where *first* list contains elements for which 
 [predicate] yielded `true`,\n * while *second* list contains elements for which [predicate] yielded `false`.\n * \n @sample samples.collections.Arrays.Transformations.partitionArrayOfPrimitives
 */
public inline fun DoubleArray.partition(predicate: (Double) -> Boolean): Pair<List<Double>, List<Double>> {
    val first = ArrayList<Double>()
    val second = ArrayList<Double>()
    for (element in this) {
        if (predicate(element)) {
            first.add(element)
        } else {
            second.add(element)
        }
    }
    return Pair(first, second)
}

/**
 * Splits the original array into pair of lists,\n * where *first* list contains elements for which 
 [predicate] yielded `true`,\n * while *second* list contains elements for which [predicate] yielded `false`.\n * \n @sample samples.collections.Arrays.Transformations.partitionArrayOfPrimitives
 */
public inline fun BooleanArray.partition(predicate: (Boolean) -> Boolean): Pair<List<Boolean>, List<Boolean>> {
    val first = ArrayList<Boolean>()
    val second = ArrayList<Boolean>()
    for (element in this) {
        if (predicate(element)) {
            first.add(element)
        } else {
            second.add(element)
        }
    }
    return Pair(first, second)
}

/**
 * Splits the original array into pair of lists,\n * where *first* list contains elements for which 
 [predicate] yielded `true`,\n * while *second* list contains elements for which [predicate] yielded `false`.\n * \n @sample samples.collections.Arrays.Transformations.partitionArrayOfPrimitives
 */
public inline fun CharArray.partition(predicate: (Char) -> Boolean): Pair<List<Char>, List<Char>> {
    val first = ArrayList<Char>()
    val second = ArrayList<Char>()
    for (element in this) {
        if (predicate(element)) {
            first.add(element)
        } else {
            second.add(element)
        }
    }
    return Pair(first, second)
}

/**
 * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \n @sample samples.collections.Operables.Operations.zipIterable\n */
public infix fun <T, R> Array<out T>.zip(other: Array<out R>): List<Pair<T, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}

/**
 * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \n @sample samples.collections.Operables.Operations.zipIterable\n */
public infix fun <R> ByteArray.zip(other: Array<out R>): List<Pair<Byte, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}

/**
 * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \n @sample samples.collections.Operables.Operations.zipIterable\n */
public infix fun <R> ShortArray.zip(other: Array<out R>): List<Pair<Short, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}

/**
 * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \n @sample samples.collections.Operables.Operations.zipIterable\n */
public infix fun <R> DoubleArray.zip(other: Array<out R>): List<Pair<Double, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}

/**
 * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \n @sample samples.collections.Operables.Operations.zipIterable\n */
public infix fun <R> CharArray.zip(other: Array<out R>): List<Pair<Char, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}
collection.

\* \* @sample samples.collections.Iterables.Operations.zipIterable
\*/
public infix fun <R>
IntArray.zip(other: Array<out R>): List<Pair<Int, R>> { un return zip(other) { t1, t2 -> t1 to t2 } un\n/\*/un \* Returns a list of pairs built from the elements of 'this' array and the [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \* @sample samples.collections.Iterables.Operations.zipIterable
\*/
LongArray.zip(other: Array<out R>): List<Pair<Long, R>> { un return zip(other) { t1, t2 -> t1 to t2 } un\n/\*/un \* Returns a list of pairs built from the elements of 'this' array and the [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \* @sample samples.collections.Iterables.Operations.zipIterable
\*/
FloatArray.zip(other: Array<out R>): List<Pair<Float, R>> { un return zip(other) { t1, t2 -> t1 to t2 } un\n/\*/un \* Returns a list of pairs built from the elements of 'this' array and the [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \* @sample samples.collections.Iterables.Operations.zipIterable
\*/
DoubleArray.zip(other: Array<out R>): List<Pair<Double, R>> { un return zip(other) { t1, t2 -> t1 to t2 } un\n/\*/un \* Returns a list of pairs built from the elements of 'this' array and the [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \* @sample samples.collections.Iterables.Operations.zipIterable
\*/
CharArray.zip(other: Array<out R>): List<Pair<Char, R>> { un return zip(other) { t1, t2 -> t1 to t2 } un\n/\*/un \* Returns a list of pairs built from the elements of 'this' array and the [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \* @sample samples.collections.Iterables.Operations.zipIterable
\*/
ByteArray.zip(other: Array<out R>, transform: (a: Byte, b: R) -> V): List<V> { un val size = minOf(size, other.size) un val list = ArrayList<V>(size) for (i in 0 until size) { un list.add(transform(this[i], other[i])) un } un return list\n\* @sample samples.collections.Iterables.Operations.zipIterableWithTransform
\*/
ShortArray.zip(other: Array<out R>, transform: (a: Short, b: R) -> V): List<V> { un val size = minOf(size, other.size) un val list = ArrayList<V>(size) for (i in 0 until size) { un list.add(transform(this[i], other[i])) un } un return list\n\* @sample samples.collections.Iterables.Operations.zipIterableWithTransform
\*/
IntArray.zip(other: Array<out R>, transform: (a: Int, b: R) -> V): List<V> { un val size = minOf(size, other.size) un val list = ArrayList<V>(size) for (i in 0 until size) { un list.add(transform(this[i], other[i])) un } un return list\n\* @sample samples.collections.Iterables.Operations.zipIterableWithTransform
\*/
LongArray.zip(other: Array<out R>, transform: (a: Long, b: R) -> V): List<V> { un val size = minOf(size, other.size) un val list = ArrayList<V>(size) for (i in 0 until size) { un list.add(transform(this[i], other[i])) un } un return list\n\* @sample samples.collections.Iterables.Operations.zipIterableWithTransform
\*/

Open Source Used In security_automation identity-guest-cs 1327
of `this` array and the other array with the same index using the provided [transform] function applied to each pair of elements.

The returned list has length of the shortest collection.

The returned list has the length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.
built from the elements of `this` collection and [other] array with the same index. The returned list has length of the shortest collection.
The returned list has length of the shortest collection. The returned list has length of the shortest collection.

```kotlin
public infix fun <R> BooleanArray.zip(other: Iterable<R>): List<Pair<Boolean, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}
```

```kotlin
public infix fun <R> CharArray.zip(other: Iterable<R>): List<Pair<Char, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}
```

```kotlin
public inline fun <T, R, V> Array<out T>.zip(other: Iterable<R>, transform: (a: T, b: R) -> V): List<V> {
    val arraySize = size
    val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))
    var i = 0
    for (element in other) {
        if (i >= arraySize) break
        list.add(transform(this[i++], element))
    }
    return list
}
```

```kotlin
public inline fun <R, V> ByteArray.zip(other: Iterable<R>, transform: (a: Byte, b: R) -> V): List<V> {
    val arraySize = size
    val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))
    var i = 0
    for (element in other) {
        if (i >= arraySize) break
        list.add(transform(this[i++], element))
    }
    return list
}
```

```kotlin
public inline fun <R, V> ShortArray.zip(other: Iterable<R>, transform: (a: Short, b: R) -> V): List<V> {
    val arraySize = size
    val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))
    var i = 0
    for (element in other) {
        if (i >= arraySize) break
        list.add(transform(this[i++], element))
    }
    return list
}
```

```kotlin
public inline fun <R, V> IntArray.zip(other: Iterable<R>, transform: (a: Int, b: R) -> V): List<V> {
    val arraySize = size
    val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))
    var i = 0
    for (element in other) {
        if (i >= arraySize) break
        list.add(transform(this[i++], element))
    }
    return list
}
```

```kotlin
    val arraySize = size
    val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))
    var i = 0
    for (element in other) {
        if (i >= arraySize) break
        list.add(transform(this[i++], element))
    }
    return list
}
```

```kotlin
public inline fun <R, V> FloatArray.zip(other: Iterable<R>, transform: (a: Float, b: R) -> V): List<V> {
    val arraySize = size
    val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))
    var i = 0
    for (element in other) {
        if (i >= arraySize) break
        list.add(transform(this[i++], element))
    }
    return list
}
```
fun <R, V> DoubleArray.zip(other: Iterable<R>, transform: (a: Double, b: R) -> V): List<V> {
    val arraySize = size
    val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))
    var i = 0
    for (element in other) {
        if (i >= arraySize) break
        list.add(transform(this[i++], element))
    }
    return list
}

/**
 * Returns a list of values built from the elements of `this` array and the [other] collection with the same index
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest collection.
 */
public inline fun <V> List.zip(other: List<V>, transform: (a: V, b: V) -> V): List<V> {
    return
}

fun <R, V> BooleanArray.zip(other: BooleanArray, transform: (a: Boolean, b: Boolean) -> V): List<V> {
    return
}

fun <R, V> CharArray.zip(other: CharArray, transform: (a: Char, b: Char) -> V): List<V> {
    return
}

fun <R, V> ByteArray.zip(other: ByteArray): List<Pair<Byte, Byte>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}

fun <R, V> ShortArray.zip(other: ShortArray): List<Pair<Short, Short>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}

fun <R, V> IntArray.zip(other: IntArray): List<Pair<Int, Int>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}

    return zip(other) { t1, t2 -> t1 to t2 }
}

fun <R, V> FloatArray.zip(other: FloatArray): List<Pair<Float, Float>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}

fun <R, V> DoubleArray.zip(other: DoubleArray): List<Pair<Double, Double>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}

fun <R, V> BooleanArray.zip(other: BooleanArray): List<Pair<Boolean, Boolean>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}

fun <R, V> CharArray.zip(other: CharArray): List<Pair<Char, Char>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}

fun <R, V> ByteArray.zip(other: ByteArray, transform: (a: Byte, b: Byte) -> V): List<V> {
    return
}

/**
 * Returns a list of values built from the elements of `this` array and the [other] collection with the same index
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest collection.
 */
public inline fun <V> List.zip(other: List<V>, transform: (a: V, b: V) -> V): List<V> {
    return
}

fun <R, V> BooleanArray.zip(other: BooleanArray, transform: (a: Boolean, b: Boolean) -> V): List<V> {
    return
}

fun <R, V> CharArray.zip(other: CharArray, transform: (a: Char, b: Char) -> V): List<V> {
    return
}

fun <R, V> ByteArray.zip(other: ByteArray): List<Pair<Byte, Byte>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}

fun <R, V> ShortArray.zip(other: ShortArray): List<Pair<Short, Short>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}

fun <R, V> IntArray.zip(other: IntArray): List<Pair<Int, Int>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}

    return zip(other) { t1, t2 -> t1 to t2 }
}

fun <R, V> FloatArray.zip(other: FloatArray): List<Pair<Float, Float>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}

fun <R, V> DoubleArray.zip(other: DoubleArray): List<Pair<Double, Double>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}

fun <R, V> BooleanArray.zip(other: BooleanArray): List<Pair<Boolean, Boolean>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}

fun <R, V> CharArray.zip(other: CharArray): List<Pair<Char, Char>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}

fun <R, V> ByteArray.zip(other: ByteArray, transform: (a: Byte, b: Byte) -> V): List<V> {
    return
}
other.size)
    val list = ArrayList<V>(size)
    for (i in 0 until size)
        list.add(transform(this[i], other[i]))
    return list

/**
 * Returns a list of values built from the elements of `this` array and the [other] array with the same index
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest array.
 */
public inline fun <V> ShortArray.zip(other: ShortArray, transform: (a: Short, b: Short) -> V): List<V> {
    val size = minOf(size, other.size)
    val list = ArrayList<V>(size)
    for (i in 0 until size)
        list.add(transform(this[i], other[i]))
    return list
}

/**
 * Returns a list of values built from the elements of `this` array and the [other] array with the same index
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest array.
 */
public inline fun <V> IntArray.zip(other: IntArray, transform: (a: Int, b: Int) -> V): List<V> {
    val size = minOf(size, other.size)
    val list = ArrayList<V>(size)
    for (i in 0 until size)
        list.add(transform(this[i], other[i]))
    return list
}

/**
 * Returns a list of values built from the elements of `this` array and the [other] array with the same index
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest array.
 */
public inline fun <V> LongArray.zip(other: LongArray, transform: (a: Long, b: Long) -> V): List<V> {
    val size = minOf(size, other.size)
    val list = ArrayList<V>(size)
    for (i in 0 until size)
        list.add(transform(this[i], other[i]))
    return list
}

/**
 * Returns a list of values built from the elements of `this` array and the [other] array with the same index
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest array.
 */
public inline fun <V> FloatArray.zip(other: FloatArray, transform: (a: Float, b: Float) -> V): List<V> {
    val size = minOf(size, other.size)
    val list = ArrayList<V>(size)
    for (i in 0 until size)
        list.add(transform(this[i], other[i]))
    return list
}

/**
 * Returns a list of values built from the elements of `this` array and the [other] array with the same index
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest array.
 */
public inline fun <V> DoubleArray.zip(other: DoubleArray, transform: (a: Double, b: Double) -> V): List<V> {
    val size = minOf(size, other.size)
    val list = ArrayList<V>(size)
    for (i in 0 until size)
        list.add(transform(this[i], other[i]))
    return list
}

/**
 * Returns a list of values built from the elements of `this` array and the [other] array with the same index
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest array.
 */
public inline fun <V> BooleanArray.zip(other: BooleanArray, transform: (a: Boolean, b: Boolean) -> V): List<V> {
    val size = minOf(size, other.size)
    val list = ArrayList<V>(size)
    for (i in 0 until size)
        list.add(transform(this[i], other[i]))
    return list
}

/**
 * Returns a list of values built from the elements of `this` array and the [other] array with the same index
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest array.
 */
public inline fun <V> CharArray.zip(other: CharArray, transform: (a: Char, b: Char) -> V): List<V> {
    val size = minOf(size, other.size)
    val list = ArrayList<V>(size)
    for (i in 0 until size)
        list.add(transform(this[i], other[i]))
    return list
}
null): A \( \{ \text{null} \} \}

var count = 0
for (element in this) \{ 
  if (++count > 1)
  buffer.append(separator)
  if (limit < 0 || count <= limit) \{ 
    buffer.appendElement(element, transform)
  } else break
\}

\( \text{null} \} \}

buffer.append(truncated)
buffer.append(postfix)
return buffer

\( \text{null} \} \}

Append the string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.\( \text{null} \} \}

* If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first \( \text{null} \} \}

* elements will be appended, followed by the [truncated] string (which defaults to "..."").\( \text{null} \} \}

* \( \text{null} \} \}

* @sample samples.collections.Collections.Transformations.joinTo

*/

public fun <A :

[limit], in which case only the first \( \text{null} \} \}

* elements will be appended, followed by the [truncated] string (which

defaults to "..."").\( \text{null} \} \}

* \( \text{null} \} \}

* @sample samples.collections.Collections.Transformations.joinTo

*/

public fun <A :
separator, prefix, postfix, limit, truncated, transform).transform().joinToString(separator, prefix, postfix, limit, truncated, transform).toString()}

/**
 * Creates a string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.
 * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit] elements will be appended, followed by the [truncated] string (which defaults to "...").
 */
public fun ShortArray.joinToString(separator: CharSequence = "", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((Short) -> CharSequence)? = null): String {
    return joinTo(StringBuilder(), separator, prefix, postfix, limit, truncated, transform).toString()}

/**
 * Creates a string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.
 * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit] elements will be appended, followed by the [truncated] string (which defaults to "...").
 */
public fun IntArray.joinToString(separator: CharSequence = "", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((Int) -> CharSequence)? = null): String {
    return joinTo(StringBuilder(), separator, prefix, postfix, limit, truncated, transform).toString()}

/**
 * Creates a string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.
 * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit] elements will be appended, followed by the [truncated] string (which defaults to "...").
 */
public fun LongArray.joinToString(separator: CharSequence = "", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((Long) -> CharSequence)? = null): String {
    return joinTo(StringBuilder(), separator, prefix, postfix, limit, truncated, transform).toString()}

/**
 * Creates a string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.
 * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit] elements will be appended, followed by the [truncated] string (which defaults to "...").
 */
public fun FloatArray.joinToString(separator: CharSequence = "", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((Float) -> CharSequence)? = null): String {
    return joinTo(StringBuilder(), separator, prefix, postfix, limit, truncated, transform).toString()}

/**
 * Creates a string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.
 * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit] elements will be appended, followed by the [truncated] string (which defaults to "...").
 */
public fun DoubleArray.joinToString(separator: CharSequence = "", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((Double) -> CharSequence)? = null): String {
    return joinTo(StringBuilder(), separator, prefix, postfix, limit, truncated, transform).toString()}

/**
 * Creates a string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.
 * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit] elements will be appended, followed by the [truncated] string (which defaults to "...").
 */
public fun BooleanArray.joinToString(separator: CharSequence = "", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((Boolean) -> CharSequence)? = null): String {
    return joinTo(StringBuilder(), separator, prefix, postfix, limit, truncated, transform).toString()}

/**
 * Creates a string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.
 * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit] elements will be appended, followed by the [truncated] string (which defaults to "...").
 */
public fun CharArray.joinToString(separator: CharSequence = "", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((Char) -> CharSequence)? = null): String {
    return joinTo(StringBuilder(), separator, prefix, postfix, limit, truncated, transform).toString()}

CharSequence = "\n\n\n...\n\n\n", transform: ((Char) -> CharSequence)? = null): String {
    return joinTo(StringBuilder(),
    separator, prefix, postfix, limit, truncated, transform).toString()
}

/**
 * Creates an 
 * [Iterable] instance that
 * wraps the original array returning its elements when being iterated.
 */
public fun <T> Array<out T>.asIterable(): Iterable<T> {
    return when (this.iterator() instanceof Iterable) { (Iterable<T>) ->
        this.iterator() // wraps the original array returning its elements when being iterated.
    }
}

/**
 * Creates an 
 * [Iterable] instance that
 * wraps the original array returning its elements when being iterated.
 */
public fun ByteArray.asIterable(): Iterable<Byte> {
    return when (this.iterator() instanceof Iterable) { (Iterable<Byte>) ->
        this.iterator() // wraps the original array returning its elements when being iterated.
    }
}

/**
 * Creates an 
 * [Iterable] instance that
 * wraps the original array returning its elements when being iterated.
 */
public fun ShortArray.asIterable(): Iterable<Short> {
    return when (this.iterator() instanceof Iterable) { (Iterable<Short>) ->
        this.iterator() // wraps the original array returning its elements when being iterated.
    }
}

/**
 * Creates an 
 * [Iterable] instance that
 * wraps the original array returning its elements when being iterated.
 */
public fun IntArray.asIterable(): Iterable<Int> {
    return when (this.iterator() instanceof Iterable) { (Iterable<Int>) ->
        this.iterator() // wraps the original array returning its elements when being iterated.
    }
}

/**
 * Creates an 
 * [Iterable] instance that
 * wraps the original array returning its elements when being iterated.
 */
public fun LongArray.asIterable(): Iterable<Long> {
    return when (this.iterator() instanceof Iterable) { (Iterable<Long>) ->
        this.iterator() // wraps the original array returning its elements when being iterated.
    }
}

/**
 * Creates an 
 * [Iterable] instance that
 * wraps the original array returning its elements when being iterated.
 */
public fun FloatArray.asIterable(): Iterable<Float> {
    return when (this.iterator() instanceof Iterable) { (Iterable<Float>) ->
        this.iterator() // wraps the original array returning its elements when being iterated.
    }
}

/**
 * Creates an 
 * [Iterable] instance that
 * wraps the original array returning its elements when being iterated.
 */
public fun DoubleArray.asIterable(): Iterable<Double> {
    return when (this.iterator() instanceof Iterable) { (Iterable<Double>) ->
        this.iterator() // wraps the original array returning its elements when being iterated.
    }
}

/**
 * Creates an 
 * [Iterable] instance that
 * wraps the original array returning its elements when being iterated.
 */
public fun BooleanArray.asIterable(): Iterable<Boolean> {
    return when (this.iterator() instanceof Iterable) { (Iterable<Boolean>) ->
        this.iterator() // wraps the original array returning its elements when being iterated.
    }
}

/**
 * Creates an 
 * [Iterable] instance that
 * wraps the original array returning its elements when being iterated.
 */
public fun CharArray.asIterable(): Iterable<Char> {
    return when (this.iterator() instanceof Iterable) { (Iterable<Char>) ->
        this.iterator() // wraps the original array returning its elements when being iterated.
    }
}
fun CharArray.asSequence(): Sequence<Char> { return when (this.iterator()) { emptyIterator() -> emptySequence() by this iterator() } }

/**
 * Returns an average value of elements in the array.
 * @param arr The array to calculate the average value of.
 * @return The average value of the elements in the array.
 */
public fun Array<out T>.average(): T {
    var sum: T = 0
    var count: Int = 0
    for (element in this) {
        sum += element
        ++count
    }
    return if (count == 0) T.NaN else sum / count
}

/**
 * Returns an average value of elements in the array.
 * @param arr The array to calculate the average value of.
 * @return The average value of the elements in the array.
 */
public fun ByteArray.average(): T {
    var sum: T = 0
    var count: Int = 0
    for (element in this) {
        sum += element
        ++count
    }
    return if (count == 0) T.NaN else sum / count
}

/**
 * Returns an average value of elements in the array.
 * @param arr The array to calculate the average value of.
 * @return The average value of the elements in the array.
 */
public fun ShortArray.average(): T {
    var sum: T = 0
    var count: Int = 0
    for (element in this) {
        sum += element
        ++count
    }
    return if (count == 0) T.NaN else sum / count
}

/**
 * Returns an average value of elements in the array.
 * @param arr The array to calculate the average value of.
 * @return The average value of the elements in the array.
 */
public fun IntArray.average(): T {
    var sum: T = 0
    var count: Int = 0
    for (element in this) {
        sum += element
        ++count
    }
    return if (count == 0) T.NaN else sum / count
}

/**
 * Returns an average value of elements in the array.
 * @param arr The array to calculate the average value of.
 * @return The average value of the elements in the array.
 */
public fun LongArray.average(): T {
    var sum: T = 0
    var count: Int = 0
    for (element in this) {
        sum += element
        ++count
    }
    return if (count == 0) T.NaN else sum / count
}

/**
 * Returns an average value of elements in the array.
 * @param arr The array to calculate the average value of.
 * @return The average value of the elements in the array.
 */
public fun FloatArray.average(): T {
    var sum: T = 0
    var count: Int = 0
    for (element in this) {
        sum += element
        ++count
    }
    return if (count == 0) T.NaN else sum / count
}

/**
 * Returns an average value of elements in the array.
 * @param arr The array to calculate the average value of.
 * @return The average value of the elements in the array.
 */
public fun DoubleArray.average(): T {
    var sum: T = 0
    var count: Int = 0
    for (element in this) {
        sum += element
        ++count
    }
    return if (count == 0) T.NaN else sum / count
}

/**
 * Returns the sum of all elements in the array.
 * @param arr The array to calculate the sum of elements in.
 * @return The sum of all elements in the array.
 */
public fun Array<out T>.sum(): T {
    var sum: T = 0
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the array.
 * @param arr The array to calculate the sum of elements in.
 * @return The sum of all elements in the array.
 */
public fun ByteArray.sum(): T {
    var sum: T = 0
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the array.
 * @param arr The array to calculate the sum of elements in.
 * @return The sum of all elements in the array.
 */
public fun ShortArray.sum(): T {
    var sum: T = 0
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the array.
 * @param arr The array to calculate the sum of elements in.
 * @return The sum of all elements in the array.
 */
public fun IntArray.sum(): T {
    var sum: T = 0
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the array.
 * @param arr The array to calculate the sum of elements in.
 * @return The sum of all elements in the array.
 */
public fun LongArray.sum(): T {
    var sum: T = 0
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the array.
 * @param arr The array to calculate the sum of elements in.
 * @return The sum of all elements in the array.
 */
public fun FloatArray.sum(): T {
    var sum: T = 0
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the array.
 * @param arr The array to calculate the sum of elements in.
 * @return The sum of all elements in the array.
 */
public fun DoubleArray.sum(): T {
    var sum: T = 0
    for (element in this) {
        sum += element
    }
    return sum
}

*/
public fun ByteArray.sum(): Int {
    var sum: Int = 0
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the array.
 */
public fun ShortArray.sum(): Int {
    var sum: Int = 0
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the array.
 */
public fun IntArray.sum(): Int {
    var sum: Int = 0
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the array.
 */
public fun LongArray.sum(): Long {
    var sum: Long = 0L
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the array.
 */
public fun FloatArray.sum(): Float {
    var sum: Float = 0.0f
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the array.
 */
public fun DoubleArray.sum(): Double {
    var sum: Double = 0.0
    for (element in this) {
        sum += element
    }
    return sum
}

@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("\RangesKt")

package kotlin.ranges

//
// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt
// See: https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib
//

import kotlin.random.*

/**
 * Returns a random element from this range.
 * 
 * @throws IllegalArgumentException if this range is empty.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun IntRange.random(): Int {
    return random(Random)
}

/**
 * Returns a random element from this range.
 * 
 * @throws IllegalArgumentException if this range is empty.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun LongRange.random(): Long {
    return random(Random)
}

/**
 * Returns a random element from this range.
 * 
 * @throws IllegalArgumentException if this range is empty.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun CharRange.random(): Char {
    return random(Random)
}

/**
 * Returns a random element from this range using the specified source of randomness.
 * 
 * @throws IllegalArgumentException if this range is empty.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@kotlin.internal.InlineOnly
public inline fun IntRange.randomOrNull(): Int? {
    return randomOrNull(Random)
}

/**
 * Returns a random element from this range using the specified source of randomness.
 * 
 * @throws IllegalArgumentException if this range is empty.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@kotlin.internal.InlineOnly
public inline fun LongRange.randomOrNull(): Long? {
    return randomOrNull(Random)
}

/**
 * Returns a random element from this range using the specified source of randomness.
 * 
 * @throws IllegalArgumentException if this range is empty.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@kotlin.internal.InlineOnly
public inline fun CharRange.randomOrNull(): Char? {
    return randomOrNull(Random)
}

/**
 * Returns a random element from this range using the specified source of randomness, or ‘null’ if this range is empty.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@kotlin.internal.InlineOnly
public inline fun IntRange.randomOrNull(random: Random): Int? {
    if (isEmpty())
        return null
    return random.nextInt(first.toInt(), last.toInt() + 1).toChar()
random.nextInt(this)

/**
 * Returns a random element from this range using the specified source of
 * randomness, or `null` if this range is empty.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public fun LongRange.randomOrNull(random: Random): Long? {
    if (isEmpty())
        return null
    return random.nextDouble(this)
}

/**
 * Returns a random element from this range using the specified source of
 * randomness, or `null` if this range is empty.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public fun CharRange.randomOrNull(random: Random): Char? {
    if (isEmpty())
        return null
    return random.nextInt(first.toInt(), last.toInt() + 1).toChar()
}

/**
 * Returns `true` if this range contains the specified 
 * [element].
 * 
 * Always returns `false` if the [element] is `null`.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline operator fun IntRange.contains(element: Int?): Boolean {
    return element != null && contains(element)
}

/**
 * Returns `true` if this range contains the specified 
 * [element].
 * 
 * Always returns `false` if the [element] is `null`.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline operator fun LongRange.contains(element: Long?): Boolean {
    return element != null && contains(element)
}

/**
 * Returns `true` if this range contains the specified 
 * [element].
 * 
 * Always returns `false` if the [element] is `null`.
 */
@SinceKotlin("1.3")
public inline operator fun CharRange.contains(element: Char?): Boolean {
    return element != null && contains(element)
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("intRangeContains")
public operator fun ClosedRange<Int>.contains(value: Byte): Boolean {
    return contains(value.toInt())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("longRangeContains")
public operator fun ClosedRange<Long>.contains(value: Byte): Boolean {
    return contains(value.toLong())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("shortRangeContains")
public operator fun ClosedRange<Short>.contains(value: Byte): Boolean {
    return contains(value.toShort())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("doubleRangeContains")
public operator fun ClosedRange<Double>.contains(value: Byte): Boolean {
    return contains(value.toDouble())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("floatRangeContains")
public operator fun ClosedRange<Float>.contains(value: Byte): Boolean {
    return contains(value.toFloat())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("intRangeContains")
public operator fun ClosedRange<Int>.contains(value: Double): Boolean {
    return value.toIntExactOrNull().let { if (it != null) contains(it) else false }
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("longRangeContains")
public operator fun ClosedRange<Long>.contains(value: Double): Boolean {
    return value.toLongExactOrNull().let { if (it != null) contains(it) else false }
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("byteRangeContains")
public operator fun ClosedRange<Byte>.contains(value: Double): Boolean {
    return value.toByteExactOrNull().let { if (it != null) contains(it) else false }
}
Checks if the specified [value] belongs to this range.`

*\/@Deprecated("This `contains` operation mixing integer and floating point arguments has ambiguous semantics and is going to be removed."")\/@DeprecatedSinceKotlin(warningSince = "1.3", errorSince = "1.4")\/@kotlin.jvm.JvmName("shortRangeContains")\)public operator fun
ClosedRange\<Short\>\.contains\(value: Double\): Boolean \{\n    return value\.toShortOrNull\(\).let \{\n        if \((it != null) contains\(it\)) else false \}\n\}*/\n
*\/@Deprecated("This `contains` operation mixing integer and floating point arguments has ambiguous semantics and is going to be removed."")\/@DeprecatedSinceKotlin(warningSince = "1.3", errorSince = "1.4")\/@kotlin.jvm.JvmName("floatRangeContains")\)public operator fun ClosedRange\<Float\>\.contains\(value: Double\): Boolean \{\n    return contains\(value\.toFloat\()\)\n\}*/

*\/@Deprecated("This `contains` operation mixing integer and floating point arguments has ambiguous semantics and is going to be removed."")\/@DeprecatedSinceKotlin(warningSince = "1.3", errorSince = "1.4")\/@kotlin.jvm.JvmName("intRangeContains")\)public operator fun ClosedRange\<Int\>\.contains\(value: Float\): Boolean \{\n    return value\.toIntOrNull\(\).let \{\n        if \((it != null) contains\(it\)) else false \}\n\}*/

*\/@Deprecated("This `contains` operation mixing integer and floating point arguments has ambiguous semantics and is going to be removed."")\/@DeprecatedSinceKotlin(warningSince = "1.3", errorSince = "1.4")\/@kotlin.jvm.JvmName("longRangeContains")\)public operator fun ClosedRange\<Long\>\.contains\(value: Float\): Boolean \{\n    return contains\(value\.toLong\()\)\n\}*/

*\/@Deprecated("This `contains` operation mixing integer and floating point arguments has ambiguous semantics and is going to be removed."")\/@DeprecatedSinceKotlin(warningSince = "1.3", errorSince = "1.4")\/@kotlin.jvm.JvmName("byteRangeContains")\)public operator fun ClosedRange\<Byte\>\.contains\(value: Float\): Boolean \{\n    return value\.toByteOrNull\(\).let \{\n        if \((it != null) contains\(it\)) else false \}\n\}*/

*\/@Deprecated("This `contains` operation mixing integer and floating point arguments has ambiguous semantics and is going to be removed."")\/@DeprecatedSinceKotlin(warningSince = "1.3", errorSince = "1.4")\/@kotlin.jvm.JvmName("shortRangeContains")\)public operator fun
ClosedRange\<Short\>\.contains\(value: Float\): Boolean \{\n    return value\.toShortOrNull\(\).let \{\n        if \((it != null) contains\(it\)) else false \}\n\}*/

*\/@Deprecated("This `contains` operation mixing integer and floating point arguments has ambiguous semantics and is going to be removed."")\/@DeprecatedSinceKotlin(warningSince = "1.3", errorSince = "1.4")\/@kotlin.jvm.JvmName("doubleRangeContains")\)public operator fun ClosedRange\<Double\>\.contains\(value: Float\): Boolean \{\n    return contains\(value\.toDouble\()\)\n\}*/
Checks if the specified \[value\] belongs to this range.

\[\text{public operator fun ClosedRange<Byte>\_.contains(value: Long): Boolean { return value.toByteExactOrNull().let { if (it != null) contains(it) else false } }\]

Checks if the specified \[value\] belongs to this range.

\[\text{public operator fun ClosedRange<Short>\_.contains(value: Long): Boolean { return value.toShortExactOrNull().let { if (it != null) contains(it) else false } }\]

Checks if the specified \[value\] belongs to this range.

\[\text{public operator fun ClosedRange<Double>\_.contains(value: Long): Boolean { return value.toDoubleExactOrNull().let { if (it != null) contains(it) else false } }\]

Checks if the specified \[value\] belongs to this range.

\[\text{public operator fun ClosedRange<Float>\_.contains(value: Long): Boolean { return value.toFloatExactOrNull().let { if (it != null) contains(it) else false } }\]

Checks if the specified \[value\] belongs to this range.

\[\text{public operator fun ClosedRange<Int>\_.contains(value: Short): Boolean { return value.toIntExactOrNull().let { if (it != null) contains(it) else false } }\]
same range with the given step. */

public infix fun IntProgression.step(step: Int): IntProgression {
    return IntProgression.fromClosedRange(first, last, if (this.step > 0) step else -step)
}

LongProgression.step(step: Long): LongProgression {
    return LongProgression.fromClosedRange(first, last, if (this.step > 0) step else -step)

    // More code here...
}

/**
 * Returns a progression that goes over the same range with the given step.
 */
public infix fun CharProgression.step(step: Int): CharProgression {
    return CharProgression.fromClosedRange(first, last, if (this.step > 0) step else -step)
}

internal fun Int.toByteExactOrNull(): Byte? {
    return if (this in Byte.MIN_VALUE.toInt()..Byte.MAX_VALUE.toInt()) this.toByte() else null
}

internal fun Long.toByteExactOrNull(): Byte? {
    return if (this in Byte.MIN_VALUE.toLong()..Byte.MAX_VALUE.toLong()) this.toByte() else null
}

internal fun Short.toByteExactOrNull(): Byte? {
    return if (this in Byte.MIN_VALUE.toShort()..Byte.MAX_VALUE.toShort()) this.toByte() else null
}

internal fun Double.toByteExactOrNull(): Byte? {
    return if (this in Byte.MIN_VALUE.toDouble()..Byte.MAX_VALUE.toDouble()) this.toInt().toByte() else null
}

internal fun Float.toByteExactOrNull(): Byte? {
    return if (this in Byte.MIN_VALUE.toFloat()..Byte.MAX_VALUE.toFloat()) this.toInt().toByte() else null
}

internal fun Long.toIntExactOrNull(): Int? {
    return if (this in Int.MIN_VALUE.toLong()..Int.MAX_VALUE.toLong()) this.toInt() else null
}

internal fun Double.toIntExactOrNull(): Int? {
    return if (this in Int.MIN_VALUE.toDouble()..Int.MAX_VALUE.toDouble()) this.toInt() else null
}

internal fun Float.toIntExactOrNull(): Int? {
    return if (this in Int.MIN_VALUE.toFloat()..Int.MAX_VALUE.toFloat()) this.toInt() else null
}

internal fun Double.toLongExactOrNull(): Long? {
    return if (this in Long.MIN_VALUE.toDouble()..Long.MAX_VALUE.toDouble()) this.toLong() else null
}

internal fun Float.toLongExactOrNull(): Long? {
    return if (this in Long.MIN_VALUE.toFloat()..Long.MAX_VALUE.toFloat()) this.toLong() else null
}

internal fun Int.toShortExactOrNull(): Short? {
    return if (this in Short.MIN_VALUE.toInt()..Short.MAX_VALUE.toInt()) this.toShort() else null
}

internal fun Long.toShortExactOrNull(): Short? {
    return if (this in Short.MIN_VALUE.toLong()..Short.MAX_VALUE.toLong()) this.toShort() else null
}

internal fun Double.toShortExactOrNull(): Short? {
    return if (this in Short.MIN_VALUE.toDouble()..Short.MAX_VALUE.toDouble()) this.toInt().toShort() else null
}

/**
 * Returns a range from this value up to but excluding the specified [to] value.
 * If the [to] value is less than or equal to `this` value, then the returned range is empty.
 */
public infix fun IntRange.until(to: Byte): IntRange {
    return this .. (to.toInt() - 1).toInt()
}

public infix fun LongRange.until(to: Long): LongRange {
    return this .. (to.long() - 1).long()
}

public infix fun CharRange.until(to: Char): CharRange {
    return this .. (to.char() - 1).char()
}

Open Source Used in security_automation identity-guest-cs 1342
the specified [to] value.

* \n * If the [to] value is less than or equal to \`this` value, then the returned range is empty.

*\npublic infix fun Long.until(to: Int): LongRange {
    \n    return this..(to.toLong() -
    1).toLong()
}

*\n\n* Returns a range from this value up to but excluding the specified [to] value.

* \n * If the [to] value is less than or equal to \`this` value, then the returned range is empty.

*\npublic infix fun Byte.until(to: Int): IntRange {
    \n    return IntRange.EMPTY
}

*\n\n* Returns a range from this value up to but excluding the specified [to] value.

* \n * If the [to] value is less than or equal to \`this` value, then the returned range is empty.

*\npublic infix fun Short.until(to: Int): IntRange {
    \n    return IntRange.EMPTY
}

*\n\n* Returns a range from this value up to but excluding the specified [to] value.

* \n * If the [to] value is less than or equal to \`this` value, then the returned range is empty.

*\npublic infix fun Int.until(to: Long): LongRange {
    \n    return this.toLong()..
    (to.toLong() - 1).toLong()
}

*\n\n* Returns a range from this value up to but excluding the specified [to] value.

* \n * If the [to] value is less than or equal to \`this` value, then the returned range is empty.

*\npublic infix fun Long.until(to: Long): LongRange {
    \n    return this.toLong()..
    (to.toLong() - 1).toLong()
}

*\n\n* Ensures that this value is not less than the specified [minimumValue].

* \n * @return this value if it's greater than or equal to the [minimumValue] or the [minimumValue] otherwise.

*\npublic fun <T : Comparable<T>>
    T.coerceAtLeast(minimumValue: T): T {
    \n    return if (this < minimumValue) minimumValue else this
}

*\n\n* Ensures that this value is not less than the specified [minimumValue].

* \n * @return this value if it's greater than or equal to the [minimumValue] or the [minimumValue] otherwise.

*\npublic fun Byte.coerceAtLeast(minimumValue: Byte): Byte {
    \n    return if (this < minimumValue) minimumValue else this
}

*\n\n* Ensures that this value is not less than the specified [minimumValue].

* \n * @return this value if it's greater than or equal to the [minimumValue] or the [minimumValue] otherwise.

*\npublic fun Short.coerceAtLeast(minimumValue: Short): Short {
    \n    return if (this < minimumValue) minimumValue else this
}

*\n\n* Ensures that this value is not less than the specified [minimumValue].

* \n * @return this value if it's greater than or equal to the [minimumValue] or the [minimumValue] otherwise.

*\npublic fun Int.coerceAtLeast(minimumValue: Int): Int {
    \n    return if (this < minimumValue) minimumValue else this
}

*\n\n* Ensures that this value is not less than the specified [minimumValue].

* \n * @return this value if it's greater than or equal to the [minimumValue] or the [minimumValue] otherwise.

*\npublic fun Long.coerceAtLeast(minimumValue: Long):


Long {\n    return if (this < minimumValue) minimumValue else this\n}\n/**\n* Ensures that this value is not less than the specified [minimumValue].\n* @return this value if it's greater than or equal to the [minimumValue] or the [minimumValue] otherwise.\n* @sample samples.comparisons.ComparableOps.coerceAtLeast\n*/ \n\npublic fun Float.coerceAtLeast(minimumValue: Float): Float {\n    \n    return if (this < minimumValue) minimumValue else this\n}\n/**\n* Ensures that this value is not less than the specified [minimumValue].\n* @return this value if it's greater than or equal to the [minimumValue] or the [minimumValue] otherwise.\n* @sample samples.comparisons.ComparableOps.coerceAtLeast\n*/ 
Double.coerceAtLeast(minimumValue: Double): Double {\n    \n    return if (this < minimumValue) minimumValue else this\n}\n/**\n* Ensures that this value is not greater than the specified [maximumValue].\n* @return this value if it's less than or equal to the [maximumValue] or the [maximumValue] otherwise.\n* @sample samples.comparisons.ComparableOps.coerceAtMost\n*/ 
Float.coerceAtMost(maximumValue: Float): Float {\n    \n    return if (this > maximumValue) maximumValue else this\n}\n/**\n* Ensures that this value is not greater than the specified [maximumValue].\n* @return this value if it's less than or equal to the [maximumValue] or the [maximumValue] otherwise.\n* @sample samples.comparisons.ComparableOps.coerceAtMost\n*/ 
Byte.coerceIn(minimumValue: Byte, maximumValue: Byte): Byte {\n    \n    return if (this < minimumValue) minimumValue else this\n}\n/**\n* Ensures that this value is not greater than the specified [maximumValue].\n* @return this value if it's less than or equal to the [maximumValue] or the [maximumValue] otherwise.\n* @sample samples.comparisons.ComparableOps.coerceIn\n*/ 
Short.coerceIn(minimumValue: Short, maximumValue: Short): Short {\n    \n    return if (this < minimumValue) minimumValue else this\n}\n/**\n* Ensures that this value is not greater than the specified [maximumValue].\n* @return this value if it's less than or equal to the [maximumValue] or the [maximumValue] otherwise.\n* @sample samples.comparisons.ComparableOps.coerceIn\n*/ 
Int.coerceIn(minimumValue: Int, maximumValue: Int): Int {\n    \n    return if (this > maximumValue) maximumValue else this\n}\n/**\n* Ensures that this value is not greater than the specified [maximumValue].\n* @return this value if it's less than or equal to the [maximumValue] or the [maximumValue] otherwise.\n* @sample samples.comparisons.ComparableOps.coerceIn\n*/ 
Long.coerceIn(minimumValue: Long, maximumValue: Long): Long {\n    \n    return if (this > maximumValue) maximumValue else this\n}\n/**\n* Ensures that this value is not greater than the specified [maximumValue].\n* @return this value if it's less than or equal to the [maximumValue] or the [maximumValue] otherwise.\n* @sample samples.comparisons.ComparableOps.coerceIn\n*/ 
Char.coerceIn(minimumValue: Char, maximumValue: Char): Char {\n    \n    return if (this > maximumValue) maximumValue else this\n}\n/**\n* Ensures that this value is not greater than the specified [maximumValue].\n* @return this value if it's less than or equal to the [maximumValue] or the [maximumValue] otherwise.\n* @sample samples.comparisons.ComparableOps.coerceIn\n*/
maximumValue: Byte): Byte {
    if (minimumValue > maximumValue) throw
    IllegalArgumentException("Cannot coerce value to an empty range: maximum $maximumValue is less than
minimum $minimumValue.")
    if (this < minimumValue) return minimumValue
    return this
}

/**
 * Ensures that this value lies in the specified range
 * \[minimumValue, maximumValue\].
 * 
 * @return this value if it's in the range, or \[minimumValue\] if this value is less than \[minimumValue\], or \[maximumValue\] if this value is greater than \[maximumValue\].
 * 
 * @sample samples.comparisons.ComparableOps.coerceIn
 */
public fun Byte.coerceIn(minimumValue: Byte, maximumValue: Byte): Byte {
    if (minimumValue > maximumValue) throw IllegalArgumentException("Cannot coerce value to an empty range: maximum $maximumValue is less than
minimum $minimumValue.")
    if (this < minimumValue) return minimumValue
    return this
}

/**
 * Ensures that this value lies in the specified range
 * \[minimumValue, maximumValue\].
 * 
 * @return this value if it's in the range, or \[minimumValue\] if this value is less than \[minimumValue\], or \[maximumValue\] if this value is greater than \[maximumValue\].
 * 
 * @sample samples.comparisons.ComparableOps.coerceIn
 */
public fun Short.coerceIn(minimumValue: Short, maximumValue: Short): Short {
    if (minimumValue > maximumValue) throw IllegalArgumentException("Cannot coerce value to an empty range: maximum $maximumValue is less than
minimum $minimumValue.")
    if (this < minimumValue) return minimumValue
    return this
}

/**
 * Ensures that this value lies in the specified range
 * \[minimumValue, maximumValue\].
 * 
 * @return this value if it's in the range, or \[minimumValue\] if this value is less than \[minimumValue\], or \[maximumValue\] if this value is greater than \[maximumValue\].
 * 
 * @sample samples.comparisons.ComparableOps.coerceIn
 */
public fun Int.coerceIn(minimumValue: Int, maximumValue: Int): Int {
    if (minimumValue > maximumValue) throw IllegalArgumentException("Cannot coerce value to an empty range: maximum $maximumValue is less than
minimum $minimumValue.")
    if (this < minimumValue) return minimumValue
    return this
}

/**
 * Ensures that this value lies in the specified range
 * \[minimumValue, maximumValue\].
 * 
 * @return this value if it's in the range, or \[minimumValue\] if this value is less than \[minimumValue\], or \[maximumValue\] if this value is greater than \[maximumValue\].
 * 
 * @sample samples.comparisons.ComparableOps.coerceIn
 */
public fun Long.coerceIn(minimumValue: Long, maximumValue: Long): Long {
    if (minimumValue > maximumValue) throw IllegalArgumentException("Cannot coerce value to an empty range: maximum $maximumValue is less than
minimum $minimumValue.")
    if (this < minimumValue) return minimumValue
    return this
}

/**
 * Ensures that this value lies in the specified range \[minimumValue, maximumValue\].
 * 
 * @return this value if it's in the range, or \[minimumValue\] if this value is less than \[minimumValue\], or \[maximumValue\] if this value is greater than \[maximumValue\].
 * 
 * @sample samples.comparisons.ComparableOps.coerceIn
 */
public fun Float.coerceIn(minimumValue: Float, maximumValue: Float): Float {
    if (minimumValue > maximumValue) throw IllegalArgumentException("Cannot coerce value to an empty range: maximum $maximumValue is less than
minimum $minimumValue.")
    if (this < minimumValue) return minimumValue
    return this
}

/**
 * Ensures that this value lies in the specified range \[minimumValue, maximumValue\].
 * 
 * @return this value if it's in the range, or \[minimumValue\] if this value is less than \[minimumValue\], or \[maximumValue\] if this value is greater than \[maximumValue\].
 * 
 * @sample samples.comparisons.ComparableOps.coerceIn
 */
public fun Double.coerceIn(minimumValue: Double, maximumValue: Double): Double {
    if (minimumValue > maximumValue) throw IllegalArgumentException("Cannot coerce value to an empty range: maximum $maximumValue is less than
minimum $minimumValue.")
    if (this < minimumValue) return minimumValue
    return this
}

/**
 * Ensures that this value lies in the specified range \[range\].
 * 
 * @return this value if it's in the range, or \[range.start\] if this value is less than \[range.start\], or \[range.endInclusive\] if this value is greater than \[range.endInclusive\].
 * 
 * @sample samples.comparisons.ComparableOps.coerceIn
 */
public fun <T : Comparable<T>> T.coerceIn(range: ClosedFloatingPointRange<T>): T {
    if (range.isEmpty()) throw IllegalArgumentException("Cannot coerce value to an empty range: $range.")
    return when {
        // this < start equiv to this <= start && !(this >= start)
        range.lessThanOrEquals(this, range.start) && !range.lessThanOrEquals(range.start, this) -> range.start
        // this > end equiv to this >= end && !(this <= end)
        range.lessThanOrEquals(range.endInclusive, this) && !range.lessThanOrEquals(this, range.endInclusive) -> range.endInclusive
        else -> this
    }
}
if this value is greater than `range.endInclusive`.

```java
public fun <T : Comparable<T>>
T.coerceIn(range: ClosedRange<T>): T {
    if (range is ClosedFloatingPointRange) {
        return this.coerceIn<T>(range)
    }
    if (range.isEmpty()) throw IllegalArgumentException("Cannot coerce value to an empty range: $range.")
    return when {
        this < range.start -> range.start
        this > range.endInclusive ->
        range.endInclusive
        else -> this
    }
}
```

```java
public fun Int.coerceIn(range: ClosedRange<Int>): Int {
    if (range is ClosedFloatingPointRange) {
        return this.coerceIn<Int>(range)
    }
    if (range.isEmpty()) throw IllegalArgumentException("Cannot coerce value to an empty range: $range.")
    return when {
        this < range.start -> range.start
        this > range.endInclusive -> range.endInclusive
        else -> this
    }
}
```

```java
public fun Long.coerceIn(range: ClosedRange<Long>): Long {
    if (range is ClosedFloatingPointRange) {
        return this.coerceIn<Long>(range)
    }
    if (range.isEmpty()) throw IllegalArgumentException("Cannot coerce value to an empty range: $range.")
    return when {
        this < range.start -> range.start
        this > range.endInclusive -> range.endInclusive
        else -> this
    }
}
```

Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.

Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

Auto-generated file. DO NOT EDIT!

```java
package kotlin

import kotlin.experimental.*

@Suppress("NON_PUBLIC_PRIMARY_CONSTRUCTOR_OF_INLINE_CLASS")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline class UByte @PublishedApi internal constructor(@PublishedApi internal val data: Byte) : Comparable<UByte> {

    /**
     * A constant holding the minimum value an instance of UByte can have.
     */
    public const val MIN_VALUE: UByte = UByte(0)

    /**
     * A constant holding the maximum value an instance of UByte can have.
     */
    public const val MAX_VALUE: UByte = UByte(-1)

    /**
     * The number of bytes used to represent an instance of UByte in a binary form.
     */
    public const val SIZE_BYTES: Int = 1

    /**
     * The number of bits used to represent an instance of UByte in a binary form.
     */
    public const val SIZE_BITS: Int = 8

    Returns zero if this value is less than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
    or a positive number if it's greater than other.
value. */
    @kotlin.internal.InlineOnly
    public inline operator fun plus(other: ULong): ULong = this.toULong().plus(other.toULong())
    /** Subtracts the other value from this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun minus(other: UByte): UInt = this.toUInt().minus(other.toUInt())
    /** Subtracts the other value from this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun minus(other: UShort): UInt = this.toUInt().minus(other.toUInt())
    /** Subtracts the other value from this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun minus(other: UInt): UInt = this.toUInt().minus(other)
    /** Subtracts the other value from this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun minus(other: ULong): ULong = this.toULong().minus(other)
    /** Multiplies this value by the other value. */
    @kotlin.internal.InlineOnly
    public inline operator fun times(other: UByte): UInt = this.toUInt().times(other.toUInt())
    /** Multiplies this value by the other value. */
    @kotlin.internal.InlineOnly
    public inline operator fun times(other: UShort): UInt = this.toUInt().times(other.toUInt())
    /** Multiplies this value by the other value. */
    @kotlin.internal.InlineOnly
    public inline operator fun times(other: UInt): UInt = this.toUInt().times(other)
    /** Multiplies this value by the other value. */
    @kotlin.internal.InlineOnly
    public inline operator fun times(other: ULong): ULong = this.toULong().times(other)
    /** Divides this value by the other value. */
    @kotlin.internal.InlineOnly
    public inline operator fun div(other: UByte): UInt = this.toUInt().div(other.toUInt())
    /** Divides this value by the other value. */
    @kotlin.internal.InlineOnly
    public inline operator fun div(other: UShort): UInt = this.toUInt().div(other.toUInt())
    /** Divides this value by the other value. */
    @kotlin.internal.InlineOnly
    public inline operator fun div(other: UInt): UInt = this.toUInt().div(other)
    /** Divides this value by the other value. */
    @kotlin.internal.InlineOnly
    public inline operator fun div(other: ULong): ULong = this.toULong().div(other)
    /** Calculates the remainder of dividing this value by the other value. */
    @kotlin.internal.InlineOnly
    public inline operator fun rem(other: UByte): UInt = this.toUInt().rem(other.toUInt())
    /** Calculates the remainder of dividing this value by the other value. */
    @kotlin.internal.InlineOnly
    public inline operator fun rem(other: UShort): UInt = this.toUInt().rem(other.toUInt())
    /** Calculates the remainder of dividing this value by the other value. */
    @kotlin.internal.InlineOnly
    public inline operator fun rem(other: UInt): UInt = this.toUInt().rem(other)
    /** Calculates the remainder of dividing this value by the other value. */
    @kotlin.internal.InlineOnly
    public inline operator fun rem(other: ULong): ULong = this.toULong().rem(other)
    /** Increments this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun inc(): UByte = UByte(data.inc())
    /** Decrements this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun dec(): UByte = UByte(data.dec())
    /** Creates a range from this value to the specified [other] value. */
    @kotlin.internal.InlineOnly
    public inline operator fun rangeTo(other: UByte): UIntRange = UIntRange(this.toUInt(), other.toUInt())
    /** Performs a bitwise AND operation between the two values. */
    @kotlin.internal.InlineOnly
    public inline infix fun and(other: UByte): UByte = UByte(this.data and other.data)
    /** Performs a bitwise OR operation between the two values. */
    @kotlin.internal.InlineOnly
    public inline infix fun or(other: UByte): UByte = UByte(this.data or other.data)
    /** Performs a bitwise XOR operation between the two values. */
    @kotlin.internal.InlineOnly
    public inline infix fun xor(other: UByte): UByte = UByte(this.data xor other.data)
    /** Inverts the bits in this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun inv(): UByte = UByte(data.inv())
    /** Converts this [UByte] value to [Byte]. */
    @kotlin.internal.InlineOnly
    public inline operator fun toByte(): Byte = data
    /** Converts this [UByte] value to [Short]. */
    @kotlin.internal.InlineOnly
    public inline operator fun toShort(): Short = data.toShort() and 0xFF
    /** Performs a bitwise OR operation between the two values. */
    @kotlin.internal.InlineOnly
    public inline operator fun or(other: UByte): UByte = UByte(data.or(other.data))
    /** Performs a bitwise XOR operation between the two values. */
    @kotlin.internal.InlineOnly
    public inline infix fun xor(other: UByte): UByte = UByte(data.xor(other.data))
    /** Calculates the remainder of dividing this value by the other value. */
    @kotlin.internal.InlineOnly
    public inline operator fun rem(other: UByte): UInt = this.toUInt().rem(other.toUInt())
    /** Calculates the remainder of dividing this value by the other value. */
    @kotlin.internal.InlineOnly
    public inline operator fun rem(other: UShort): UInt = this.toUInt().rem(other.toUInt())
    /** Calculates the remainder of dividing this value by the other value. */
    @kotlin.internal.InlineOnly
    public inline operator fun rem(other: UInt): UInt = this.toUInt().rem(other)
    /** Calculates the remainder of dividing this value by the other value. */
    @kotlin.internal.InlineOnly
    public inline operator fun rem(other: ULong): ULong = this.toULong().rem(other)
    /** Creates a range from this value to the specified [other] value. */
    @kotlin.internal.InlineOnly
    public inline operator fun rangeTo(other: UByte): UIntRange = UIntRange(this.toUInt(), other.toUInt())
    /** Performs a bitwise AND operation between the two values. */
    @kotlin.internal.InlineOnly
    public inline infix fun and(other: UByte): UByte = UByte(this.data and other.data)
    /** Performs a bitwise OR operation between the two values. */
    @kotlin.internal.InlineOnly
    public inline infix fun or(other: UByte): UByte = UByte(this.data or other.data)
    /** Performs a bitwise XOR operation between the two values. */
    @kotlin.internal.InlineOnly
    public inline infix fun xor(other: UByte): UByte = UByte(this.data xor other.data)
value, whereas the most significant 24 bits are filled with zeros.

* The resulting `Long` value represents the same numerical value as this `UByte`. * The least significant 8 bits of the resulting `Long` value are the same as the bits of this `UByte` value, whereas the most significant 56 bits are filled with zeros.

* The resulting `UShort` value represents the same numerical value as this `UByte`. * The least significant 8 bits of the resulting `UShort` value are the same as the bits of this `UByte`, whereas the most significant 8 bits are filled with zeros.

* The most significant 24 bits are filled with zeros.

* The resulting `UShort` value represents the same numerical value as this `UByte`. * The least significant 8 bits of the resulting `UShort` value are the same as the bits of this `UByte`, whereas the most significant 56 bits are filled with zeros.

* The least significant 8 bits of the resulting `ULong` value are the same as the bits of this `UByte` value, whereas the most significant 56 bits are filled with zeros.

* The least significant 8 bits of the resulting `ULong` value are the same as the bits of this `UByte`, whereas the most significant 56 bits are filled with zeros.

* The least significant 8 bits of the resulting `ULong` value are the same as the bits of this `UByte`, whereas the most significant 56 bits are filled with zeros.

* The least significant 8 bits of the resulting `ULong` value are the same as the bits of this `UByte`, whereas the most significant 56 bits are filled with zeros.

* The least significant 8 bits of the resulting `ULong` value are the same as the bits of this `UByte`, whereas the most significant 56 bits are filled with zeros.

* The least significant 8 bits of the resulting `ULong` value are the same as the bits of this `UByte`, whereas the most significant 56 bits are filled with zeros.

* The least significant 8 bits of the resulting `ULong` value are the same as the bits of this `UByte`, whereas the most significant 56 bits are filled with zeros.

* The least significant 8 bits of the resulting `ULong` value are the same as the bits of this `UByte`, whereas the most significant 56 bits are filled with zeros.

* The least significant 8 bits of the resulting `ULong` value are the same as the bits of this `UByte`, whereas the most significant 56 bits are filled with zeros.

* The least significant 8 bits of the resulting `ULong` value are the same as the bits of this `UByte`, whereas the most significant 56 bits are filled with zeros.

* The least significant 8 bits of the resulting `ULong` value are the same as the bits of this `UByte`, whereas the most significant 56 bits are filled with zeros.

* The least significant 8 bits of the resulting `ULong` value are the same as the bits of this `UByte`, whereas the most significant 56 bits are filled with zeros.

* The least significant 8 bits of the resulting `ULong` value are the same as the bits of this `UByte`, whereas the most significant 56 bits are filled with zeros.

* The least significant 8 bits of the resulting `ULong` value are the same as the bits of this `UByte`, whereas the most significant 56 bits are filled with zeros.

* The least significant 8 bits of the resulting `ULong` value are the same as the bits of this `UByte`, whereas the most significant 56 bits are filled with zeros.

* The least significant 8 bits of the resulting `ULong` value are the same as the bits of this `UByte`, whereas the most significant 56 bits are filled with zeros.

* The least significant 8 bits of the resulting `ULong` value are the same as the bits of this `UByte`, whereas the most significant 56 bits are filled with zeros.

* The least significant 8 bits of the resulting `ULong` value are the same as the bits of this `UByte`, whereas the most significant 56 bits are filled with zeros.

* The least significant 8 bits of the resulting `ULong` value are the same as the bits of this `UByte`, whereas the most significant 56 bits are filled with zeros.

* The least significant 8 bits of the resulting `ULong` value are the same as the bits of this `UByte`, whereas the most significant 56 bits are filled with zeros.

* The least significant 8 bits of the resulting `ULong` value are the same as the bits of this `UByte`, whereas the most significant 56 bits are filled with zeros.

* The least significant 8 bits of the resulting `ULong` value are the same as the bits of this `UByte`, whereas the most significant 56 bits are filled with zeros.

* The least significant 8 bits of the resulting `ULong` value are the same as the bits of this `UByte`, whereas the most significant 56 bits are filled with zeros.

* The least significant 8 bits of the resulting `ULong` value are the same as the bits of this `UByte`, whereas the most significant 56 bits are filled with zeros.

* The least significant 8 bits of the resulting `ULong` value are the same as the bits of this `UByte`, whereas the most significant 56 bits are filled with zeros.

* The least significant 8 bits of the resulting `ULong` value are the same as the bits of this `UByte`, whereas the most significant 56 bits are filled with zeros.

* The least significant 8 bits of the resulting `ULong` value are the same as the bits of this `UByte`, whereas the most significant 56 bits are filled with zeros.
MIN_VALUE: UInt = UInt(0)

public const val MAX_VALUE: UInt = UInt(-1)

public const val SIZE_BYTES: Int = 32

public const val SIZE_BITS: Int = 32

/**
 * The number of bytes used to represent an instance of UInt in a binary form.
 */

/**
 * The number of bits used to represent an instance of UInt in a binary form.
 */

/**
 * A constant holding the maximum value an instance of UInt can have.
 */

@kotlin.internal.InlineOnly
public inline operator fun compareTo(other: UByte): Int = this.compareTo(other.toUInt())

@kotlin.internal.InlineOnly
public inline operator fun compareTo(other: UShort): Int = this.compareTo(other.toUInt())

@kotlin.internal.InlineOnly
public inline operator fun compareTo(other: UInt): Int = uintCompare(this.data, other.data)

@kotlin.internal.InlineOnly
public inline operator fun compareTo(other: ULong): Int = this.toULong().compareTo(other)

/**
 * Compares this value with the specified value for order.
 * Returns zero if this value is equal to the specified other value, a negative number if it's less than other,
 * or a positive number if it's greater than other.
 */

@kotlin.internal.InlineOnly
public inline operator fun compareTo(other: UByte): Int = this.compareTo(other.toUInt())

@kotlin.internal.InlineOnly
public inline operator fun compareTo(other: UShort): Int = this.compareTo(other.toUInt())

@kotlin.internal.InlineOnly
public inline operator fun compareTo(other: UInt): UInt = UInt(this.data.compareTo(other.data))

@kotlin.internal.InlineOnly
public inline operator fun compareTo(other: ULong): ULong = this.toULong().compareTo(other)

/**
 * Compares this value with the specified value for order.
 * Returns zero if this value is equal to the specified other value, a negative number if it's less than other,
 * or a positive number if it's greater than other.
 */

@kotlin.internal.InlineOnly
public inline operator fun compareTo(other: UByte): Int = this.compareTo(other.toUInt())

@kotlin.internal.InlineOnly
public inline operator fun compareTo(other: UShort): Int = this.compareTo(other.toUInt())

@kotlin.internal.InlineOnly
public inline operator fun compareTo(other: UInt): UInt = UInt(this.data.compareTo(other.data))

@kotlin.internal.InlineOnly
public inline operator fun compareTo(other: ULong): ULong = this.toULong().compareTo(other)

/**
 * Compares this value with the specified value for order.
 * Returns zero if this value is equal to the specified other value, a negative number if it's less than other,
 * or a positive number if it's greater than other.
 */

@kotlin.internal.InlineOnly
public inline operator fun compareTo(other: UByte): Int = this.compareTo(other.toUInt())

@kotlin.internal.InlineOnly
public inline operator fun compareTo(other: UShort): Int = this.compareTo(other.toUInt())

@kotlin.internal.InlineOnly
public inline operator fun compareTo(other: UInt): UInt = UInt(this.data.compareTo(other.data))

@kotlin.internal.InlineOnly
public inline operator fun compareTo(other: ULong): ULong = this.toULong().compareTo(other)

/**
 * Compares this value with the specified value for order.
 * Returns zero if this value is equal to the specified other value, a negative number if it's less than other,
 * or a positive number if it's greater than other.
 */

@kotlin.internal.InlineOnly
public inline operator fun compareTo(other: UByte): Int = this.compareTo(other.toUInt())

@kotlin.internal.InlineOnly
public inline operator fun compareTo(other: UShort): Int = this.compareTo(other.toUInt())

@kotlin.internal.InlineOnly
public inline operator fun compareTo(other: UInt): UInt = UInt(this.data.compareTo(other.data))

@kotlin.internal.InlineOnly
public inline operator fun compareTo(other: ULong): ULong = this.toULong().compareTo(other)

/**
 * Compares this value with the specified value for order.
 * Returns zero if this value is equal to the specified other value, a negative number if it's less than other,
 * or a positive number if it's greater than other.
 */

@kotlin.internal.InlineOnly
public inline operator fun compareTo(other: UByte): Int = this.compareTo(other.toUInt())

@kotlin.internal.InlineOnly
public inline operator fun compareTo(other: UShort): Int = this.compareTo(other.toUInt())

@kotlin.internal.InlineOnly
public inline operator fun compareTo(other: UInt): UInt = UInt(this.data.compareTo(other.data))

@kotlin.internal.InlineOnly
public inline operator fun compareTo(other: ULong): ULong = this.toULong().compareTo(other)
@kotlin.internal.InlineOnly
public inline operator fun rem(other: ULong): ULong =
this.toULong().rem(other)/*n
/** Increments this value. */
@kotlin.internal.InlineOnly
public inline operator fun inc(): UInt = UInt(data.inc())/*n
/** Decrements this value. */
@kotlin.internal.InlineOnly
public inline operator fun dec(): UInt = UInt(data.dec())/*n
/** Creates a range from this value to the specified
[other] value. */
@kotlin.internal.InlineOnly
public inline fun rangeTo(other: UInt): UIntRange =
UIntRange(this, other)/*n
/** Shifts this value left by the [bitCount] number of bits. */
* Note that only the five lowest-order bits of the [bitCount] are used as the shift distance./*n
* The shift distance actually used is therefore always in the range 0..31./*n
* The resulting `Byte` value represents the same numerical value as this `UInt`./*n
* The resulting `Byte` value is represented by the least significant 8 bits of this `UInt` value./*n
* Note that the resulting `Byte` value may be negative./*n
* Converts this [UInt] value to [Byte]./*n
* If this value is less than or equals to [Byte.MAX_VALUE], the resulting `Byte` value represents
* the same numerical value as this `UInt`./*n
* The resulting `Byte` value is represented by the least significant 8 bits of this `UInt` value./*n
* Note that the resulting `Byte` value may be negative./*n
* Converts this [UInt] value to [Short]./*n
* If this value is less than or equals to [Short.MAX_VALUE], the resulting `Short` value represents
* the same numerical value as this `UInt`./*n
* The resulting `Short` value is represented by the least significant 16 bits of this `UInt` value./*n
* Note that the resulting `Short` value may be negative./*n
* Converts this [UInt] value to [Int]./*n
* If this value is less than or equals to [Int.MAX_VALUE], the resulting `Int` value represents
* the same numerical value as this `UInt`. Otherwise the result is negative./*n
* The resulting `Int` value has the same binary representation as this `UInt` value./*n
* Converts this [UInt] value to [Long]./*n
* The resulting `Long` value represents the same numerical value as this `UInt`./*n
* The least significant 32 bits of the resulting `Long` value are the same as the bits of this `UInt` value./*n
* Whereas the most significant 32 bits are filled with zeros./*n
* Converts this [UInt] value to [UBYTE]./*n
* If this value is less than or equals to [UBYTE.MAX_VALUE], the resulting `UBYTE` value represents
* the same numerical value as this `UInt`./*n
* The resulting `UBYTE` value is represented by the least significant 8 bits of this `UInt` value./*n
* Converts this [UInt] value to [USHORT]./*n
* If this value is less than or equals to [USHORT.MAX_VALUE], the resulting `USHORT` value represents
* the same numerical value as this `UInt`./*n
* The resulting `USHORT` value is represented by the least significant 16 bits of this `UInt` value./*n
* Converts this [UInt] value to [UCHAR]./*n
* If this value is less than or equals to [UCHAR.MAX_VALUE], the resulting `UCHAR` value represents
* the same numerical value as this `UInt`. /*n
* The least significant 32 bits of the resulting `UCHAR` value are the same as the bits of this `UInt` value./*n
* Whereas the most significant 32 bits are filled with zeros./*n
* Converts this [UInt] value to [FLOAT]./*n
* The resulting value is the closest `Float` to this `UInt` value. In case when this `UInt` value is exactly
between two `Float`s, the one with zero at least significant bit of mantissa is selected.*/

Open Source Used in security_automation identity-guest-cs 1350
public inline fun toFloat(): Float = this.toDouble().toFloat()
    /**
     * Converts this [UInt] value to [Double].
     * The resulting `Double` value represents the same numerical value as this `UInt`. * *
     * If this value is positive, the resulting `UInt` value represents the same numerical value as this `Byte`
     * whereas the most significant 24 bits are filled with the sign bit of this value.
     */
    @kotlin.internal.InlineOnly
    public inline fun toDouble(): Double = uintToDouble(data)

    public override fun toString(): String = toLong().toString()

/**
 * Attempts to convert this `Byte` value to `UInt`. If this value is positive, the resulting `UInt` value
 * represents the same numerical value as this `Byte`. `NaN` is returned if this `Byte` value is negative or
 * `NaN`, [UInt.MAX_VALUE] if it's bigger than [UInt.MAX_VALUE].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun Byte.toUInt(): UInt = UInt(this.toInt())
/**
 * Attempts to convert this `Short` value to `UInt`. If this value is positive, the resulting `UInt` value
 * represents the same numerical value as this `Short`. `NaN` is returned if this `Short` value is negative or
 * `NaN`, [UInt.MAX_VALUE] if it's bigger than [UInt.MAX_VALUE].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun Short.toUInt(): UInt = UInt(this.toInt())
/**
 * Attempts to convert this `Int` value to `UInt`. If this value is positive, the resulting `UInt` value
 * represents the same numerical value as this `Int`. `NaN` is returned if this `Int` value is negative or
 * `NaN`, [UInt.MAX_VALUE] if it's bigger than [UInt.MAX_VALUE].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun Int.toUInt(): UInt = UInt(this)
/**
 * Attempts to convert this `Long` value to `UInt`. If this value is positive and less than or equals to
 * [UInt.MAX_VALUE], the resulting `UInt` value represents the same numerical value as this `Int`. `NaN` is returned if this `Int` value is negative or
 * `NaN`, [UInt.MAX_VALUE] if it's bigger than [UInt.MAX_VALUE].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun Long.toUInt(): UInt = UInt(this.toInt())
/**
 * Attempts to convert this `Float` value to `UInt`. If this value is positive and less than or equals to
 * [UInt.MAX_VALUE], the resulting `UInt` value represents the same numerical value as this `Int`. `NaN` is returned if this `Int` value is negative or
 * `NaN`, [UInt.MAX_VALUE] if it's bigger than [UInt.MAX_VALUE].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun Float.toUInt(): UInt = doubleToUInt(this.toDouble())
/**
 * Attempts to convert this `Double` value to `UInt`. If this value is positive and less than or equals to
 * [UInt.MAX_VALUE], the resulting `UInt` value represents the same numerical value as this `Int`. `NaN` is returned if this `Int` value is negative or
 * `NaN`, [UInt.MAX_VALUE] if it's bigger than [UInt.MAX_VALUE].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun Double.toUInt(): UInt = doubleToUInt(this)
specifies value for order. * Returns zero if this value is equal to the specified other value, a negative number if it's less than other, * or a positive number if it's greater than other. *%n *%n @kotlin.internal.InlineOnly
public inline operator fun compareTo(other: UInt): Int = this.toUInt().compareTo(other) /%/n *%n *%n Compares this value with the specified value for order. *%n *%n Returns zero if this value is equal to the specified other value, a negative number if it's less than other, * or a positive number if it's greater than other. *%n *%n @kotlin.internal.InlineOnly
public inline operator fun compareTo(other: ULong): Int = this.toULong().compareTo(other) /%/n Adds the other value to this value. *%n @kotlin.internal.InlineOnly
public inline operator fun plus(other: UByte): UInt = this.toUInt().plus(other.toUInt()) /%n /* Adds the other value to this value. *%n @kotlin.internal.InlineOnly
public inline operator fun plus(other: UShort): UInt = this.toUInt().plus(other.toUInt()) /%n /* Adds the other value to this value. *%n @kotlin.internal.InlineOnly
public inline operator fun plus(other: UInt): UInt = this.toUInt().plus(other) /%n /* Adds the other value to this value. *%n @kotlin.internal.InlineOnly
public inline operator fun plus(other: ULong): ULong = this.toULong().plus(other) /%n /* Adds the other value from this value. *%n @kotlin.internal.InlineOnly
public inline operator fun minus(other: UByte): UInt = this.toUInt().minus(other.toUInt()) /%n /* Subtracts the other value from this value. *%n @kotlin.internal.InlineOnly
public inline operator fun minus(other: UShort): UInt = this.toUInt().minus(other.toUInt()) /%n /* Subtracts the other value from this value. *%n @kotlin.internal.InlineOnly
public inline operator fun minus(other: UInt): UInt = this.toUInt().minus(other) /%n /* Subtracts the other value from this value. *%n @kotlin.internal.InlineOnly
public inline operator fun minus(other: ULong): ULong = this.toULong().minus(other) /%n /* Multiplies this value by the other value. *%n @kotlin.internal.InlineOnly
public inline operator fun times(other: UByte): UInt = this.toUInt().times(other.toUInt()) /%n /* Multiplies this value by the other value. *%n @kotlin.internal.InlineOnly
public inline operator fun times(other: UShort): UInt = this.toUInt().times(other.toUInt()) /%n /* Multiplies this value by the other value. *%n @kotlin.internal.InlineOnly
public inline operator fun times(other: UInt): UInt = this.toUInt().times(other) /%n /* Multiplies this value by the other value. *%n @kotlin.internal.InlineOnly
public inline operator fun times(other: ULong): ULong = this.toULong().times(other) /%n /* Divides this value by the other value. *%n @kotlin.internal.InlineOnly
public inline operator fun div(other: UByte): UInt = this.toUInt().div(other.toUInt()) /%n /* Divides this value by the other value. *%n @kotlin.internal.InlineOnly
public inline operator fun div(other: UShort): UInt = this.toUInt().div(other.toUInt()) /%n /* Divides this value by the other value. *%n @kotlin.internal.InlineOnly
public inline operator fun div(other: UInt): UInt = this.toUInt().div(other) /%n /* Divides this value by the other value. *%n @kotlin.internal.InlineOnly
public inline operator fun div(other: ULong): ULong = this.toULong().div(other) /%n /* Calculates the remainder of dividing this value by the other value. *%n @kotlin.internal.InlineOnly
public inline operator fun rem(other: UByte): UInt = this.toUInt().rem(other.toUInt()) /%n /* Calculates the remainder of dividing this value by the other value. *%n @kotlin.internal.InlineOnly
public inline operator fun rem(other: UShort): UInt = this.toUInt().rem(other.toUInt()) /%n /* Calculates the remainder of dividing this value by the other value. *%n @kotlin.internal.InlineOnly
public inline operator fun rem(other: UInt): UInt = this.toUInt().rem(other) /%n /* Calculates the remainder of dividing this value by the other value. *%n @kotlin.internal.InlineOnly
public inline operator fun rem(other: ULong): ULong = this.toULong().rem(other) /%n /* Increments this value. *%n @kotlin.internal.InlineOnly
public inline operator fun inc(): UShort = UShort(data.inc()) /%n /* Decrements this value. *%n @kotlin.internal.InlineOnly
public inline operator fun dec(): UShort = UShort(data.dec()) /%n /* Creates a range from this value to the specified [other] value. *%n @kotlin.internal.InlineOnly
public inline operator fun rangeTo(other: UShort): UIntRange = UIntRange(this.toUInt(), other.toUInt()) /%n /* Performs a bitwise AND operation between the two values. *%n @kotlin.internal.InlineOnly
public inline operator fun and(other: UShort): UShort = UShort(this.data and other.data) /%n /* Performs a bitwise OR operation between the two values. *%n @kotlin.internal.InlineOnly
public inline operator fun or(other: UShort): UShort = UShort(this.data or other.data) /%n /* Performs a bitwise XOR operation between the two values. *%n @kotlin.internal.InlineOnly
public inline operator fun xor(other: UShort): UShort = UShort(this.data xor other.data) /%n /* Performs a bitwise XOR operation between the two values. *%n @kotlin.internal.InlineOnly
public inline operator fun xor(other: UShort): UShort = UShort(this.data xor other.data) /%n /* Performs a bitwise XOR operation between the two values. *%n @kotlin.internal.InlineOnly
public inline operator fun xor(other: UShort): UShort = UShort(this.data xor other.data)
other.data)

    public override fun toString(): String = toInt().toString()
this value is positive and less than or equals to [UShort.MAX_VALUE], the resulting `UShort` value represents the same numerical value as this `Long`. The resulting `UShort` value is represented by the least significant 16 bits of this `Long` value.

```
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun Long.toUShort(): UShort = UShort(this.toShort())
```

```
/*
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

// Auto-generated file. DO NOT EDIT!

package kotlin.ranges

/**
 * A range of values of type `Char`. *
 */
public class CharRange(start: Char, endInclusive: Char) : CharProgression(start, endInclusive, 1), ClosedRange<Char> {
    override val start: Char get() = first
    override val endInclusive: Char get() = last

    override fun contains(value: Char): Boolean = first <= value && value <= last
    override fun override fun contains(value: Int): Boolean = first <= value && value <= last
    override fun override fun equals(other: Any?): Boolean = other is CharRange && (isEmpty() && other.isEmpty() || first == other.first && last == other.last)

    override fun override fun hashCode(): Int = if (isEmpty()) -1 else (31 * first.toInt() + last.toInt())

    override fun override fun toString(): String = "$first..$last"

    companion object {
        /** An empty range of values of type Char. */
        public val EMPTY: CharRange = CharRange(1.toChar(), 0.toChar())
    }
}

/**
 * A range of values of type `Int`.
 */
public class IntRange(start: Int, endInclusive: Int) : IntProgression(start, endInclusive, 1), ClosedRange<Int> {
    override val start: Int get() = first
    override val endInclusive: Int get() = last

    override fun contains(value: Int): Boolean = first <= value && value <= last
    override fun override fun equals(other: Any?): Boolean = other is IntRange && (isEmpty() && other.isEmpty() || first == other.first && last == other.last)

    override fun override fun hashCode(): Int = if (isEmpty()) -1 else (31 * first + last)

    override fun override fun toString(): String = "$first..$last"

    companion object {
        /** An empty range of values of type Int. */
        public val EMPTY: IntRange = IntRange(1, 0)
    }
}

/**
 * A range of values of type `Long`.
 */
public class LongRange(start: Long, endInclusive: Long) : LongProgression(start, endInclusive, 1), ClosedRange<Long> {
    override val start: Long get() = first
    override val endInclusive: Long get() = last

    override fun contains(value: Long): Boolean = first <= value && value <= last
    override fun override fun equals(other: Any?): Boolean = other is LongRange && (isEmpty() && other.isEmpty() || first == other.first && last == other.last)

    override fun override fun hashCode(): Int = if (isEmpty()) -1 else (31 * (first xor (first ushr 32)) + (last xor (last ushr 32))).toInt()

    override fun override fun toString(): String = "$first..$last"

    companion object {
        /** An empty range of values of type Long. */
        public val EMPTY: LongRange = LongRange(1, 0)
    }
}

// Auto-generated file. DO NOT EDIT!

package kotlin.collections

import kotlin.contracts.*
import kotlin.random.Random

internal object EmptyIterator : ListIterator<Nothing> {
    override fun hasNext(): Boolean = false
    override fun hasPrevious(): Boolean = false
    override fun nextIndex(): Int = 0
    override fun previousIndex(): Int = -1
    override fun next(): Nothing = throw NoSuchElementException()
    override fun previous(): Nothing = throw NoSuchElementException()
}

internal object EmptyList : List<Nothing>, Serializable, RandomAccess {
    private const val serialVersionUID: Long = -7390468764508069838L

    override fun equals(other: Any?): Boolean = other is List<*> && other.isEmpty()
    override fun override fun hashCode(): Int = 1
    override fun override fun toString(): String = "[]"

    override val size: Int get() = 0
    override fun override fun contains(element: Nothing): Boolean = false
    override fun override fun containsAll(elements: Collection<Nothing>): Boolean = elements.isEmpty()

    override fun override fun get(index: Int): Nothing = throw IndexOutOfBoundsException("Empty list doesn't contain element at index $index.")
    override fun override fun override fun put(index: Int, element: Nothing): Nothing = throw IndexOutOfBoundsException("Empty list doesn't contain element at index $index.")
    override fun override fun containsList(list: Collection<Nothing>): Boolean = throw IndexOutOfBoundsException("Empty list doesn't contain list at index $index.")
    override fun override fun getLists(): Collection<Nothing> = throw IndexOutOfBoundsException("Empty list doesn't contain lists.")

Open Source Used In security_automation identity-guest-cs 1354
subList(fromIndex: Int, toIndex: Int): List<Nothing> {
    if (fromIndex == 0 && toIndex == 0) return this
    throw IndexOutOfBoundsException("fromIndex: $fromIndex, toIndex: $toIndex")
}

private fun readResolve(): Any = EmptyList
}

internal fun <T> Array<out T>.asCollection(): Collection<T> =
    ArrayAsCollection(this, isVarargs = false)

private class ArrayAsCollection<T>(val values: Array<out T>, val isVarargs: Boolean) : Collection<T> {
    override val size: Int get() = values.size
    override fun contains(element: T): Boolean = values.contains(element)
    override fun containsAll(elements: Collection<T>): Boolean = elements.all { contains(it) }
    override fun iterator(): Iterator<T> = values.iterator()
    // override hidden toArray implementation to prevent copying of values array
    public fun toArray(): Array<out Any?> = values.copyToArrayOfAny(isVarargs)
}

/**
 * Returns an empty read-only list. The returned list is serializable (JVM).
 * @sample samples.collections.Collections.Lists.emptyList
 */
public fun <T> emptyList(): List<T> = EmptyList

/**
 * Returns a new read-only list of given elements. The returned list is serializable (JVM).
 * @sample samples.collections.Collections.Lists.readOnlyList
 */
public fun <T> listOf(vararg elements: T): List<T> =
    if (elements.size > 0) elements.asList() else emptyList()

/**
 * Returns an empty new [MutableList].
 * @sample samples.collections.Collections.Lists.mutableList
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public inline fun <T> mutableListOf(): MutableList<T> =
    ArrayList()

/**
 * Returns a new [Array] with the given elements.
 * @sample samples.collections.Collections.Lists.arrayList
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public inline fun <T> arrayListOf(vararg elements: T): ArrayList<T> =
    if (elements.size == 0) ArrayList() else ArrayList(ArrayAsCollection(elements, isVarargs = true))

/**
 * Creates a new read-only list either of single given element, if it is not null, or empty list if the element is null. The returned list is serializable (JVM).
 * @sample samples.collections.Collections.Lists.emptyList
 */
public fun <T : Any> listOfNotNull(element: T?): List<T> = if (element != null) listOf(element) else emptyList()

/**
 * Returns a new read-only list only of those given elements, that are not null. The returned list is serializable (JVM).
 * @sample samples.collections.Collections.Lists.emptyList
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
public inline fun <E>
buildList(@BuilderInference builderAction: MutableList<E>.() -> Unit): List<E> {
    contract {
        callsInPlace(builderAction, InvocationKind.EXACTLY_ONCE)
    } return buildListInternal(builderAction)
}

buildListInternal(builderAction): Int -> Unit: List<Int> {
    contract {
        callsInPlace(builderAction, InvocationKind.EXACTLY_ONCE)
    } return buildListInternal(builderAction)
the list within the specified range;\n * otherwise, the inverted insertion point `(-insertion point - 1)`.
 * The insertion point is defined as the index at which the element should be inserted;\n * so that the list (or the specified subrange of list) still remains sorted.\n * @sample
samples.collections.Collections.Lists.binarySearchOnComparable\n * @sample
samples.collections.Collections.Lists.binarySearchWithBoundaries\n */\npublic fun \(<T : Comparable<T>>\nList\<T?>.binarySearch\(element: T?, \fromIndex: Int = 0, \toIndex: Int = size): Int \{ rangeCheck\(size, \fromIndex, \toIndex\)\n  var low = fromIndex\n  var high = toIndex - 1\n  while \(low <= high\) \{
    val mid = (low + high).ushr(1) // safe from overflows\n    val midVal = get\(mid\)\n    val cmp = compareValues\(midVal, element\)\n    if \(cmp < 0\)\n      low = mid + 1\n    else if \(cmp > 0\)\n      high = mid - 1\n    else return mid \// key found\n  \}
  \return return -\(low + 1\) \// key not found\n\} \* Searches this list or its range for the provided [element] using the binary search algorithm.\n * The list is expected to be sorted into ascending order according to the specified [comparator].\n * otherwise the result is undefined.\n * If the list contains multiple elements equal to the specified [element], there is no guarantee which one will be found.\n *\n * *null* value is considered to be less than any non-null value.\n *\n * @return the index of the element, if it is contained in the list within the specified range;\n * otherwise, the inverted insertion point `(-insertion point - 1)`.
 * The insertion point is defined as the index at which the element should be inserted,\n * so that the list (or the specified subrange of list) still remains sorted according to the specified [comparator].\n * @sample
samples.collections.Collections.Lists.binarySearchWithComparator\n */\npublic fun \(<T\>
List\<T>.binarySearch\(element: T, \comparator: Comparator<in T>, \fromIndex: Int = 0, \toIndex: Int = size\): Int \{ rangeCheck\(size, \fromIndex, \toIndex\)\n  var low = fromIndex\n  var high = toIndex - 1\n  while \(low <= high\) \{
    val mid = (low + high).ushr(1) // safe from overflows\n    val midVal = get\(mid\)\n    val cmp = comparator.compare\(midVal, element\)\n    if \(cmp < 0\)\n      low = mid + 1\n    else if \(cmp > 0\)\n      high = mid - 1\n    else return mid \// key found\n  \}
  \return return -\(low + 1\) \// key not found\n\} \* Searches this list or its range for an element having the key returned by the specified [selector] function equal to the provided [key] value using the binary search algorithm.\n * The list is expected to be sorted into ascending order according to the Comparable natural ordering of keys of its elements.\n * otherwise the result is undefined.\n * If the list contains multiple elements with the specified [key], there is no guarantee which one will be found.\n *\n * *null* value is considered to be less than any non-null value.\n *\n * @return the index of the specified element, if it is contained in the list within the specified range;\n * otherwise, the inverted insertion point `(-insertion point - 1)`.
 * The insertion point is defined as the index at which the element should be inserted,\n * so that the list (or the specified subrange of list) still remains sorted.\n * @sample
samples.collections.Collections.Lists.binarySearchByKey\n */\npublic inline fun \(<T, K : Comparable<K>>\nList\<T>.binarySearch\(key: K?, \fromIndex: Int = 0, \toIndex: Int = size\, crossinline selector: (T) -> K\): Int \{ rangeCheck\(size, \fromIndex, \toIndex\)\n  var low = fromIndex\n  var high = toIndex - 1\n  while \(low <= high\) \{
    val mid = (low + high).ushr(1) // safe from overflows\n    val midVal = get\(mid\)\n    val cmp = comparator.compare\(selector\(it\), key\)\n    if \(cmp < 0\)\n      low = mid + 1\n    else if \(cmp > 0\)\n      high = mid - 1\n    else return mid \// key found\n  \}
  \return return -\(low + 1\) \// key not found\n\} \* Searches this list or its range for an element for which the given [comparison] function returns zero using the binary search algorithm.\n *\n * The list is expected to be sorted so that the signs of the [comparison] function's return values ascend on the list elements;\n * i.e. negative values come before zero and zeroes come before positive values.\n * Otherwise, the result is undefined.\n *\n * If the list contains multiple elements for which [comparison] returns zero, there is no guarantee which one will be found.\n *\n * @param comparison function that returns zero when called on the list element being searched.\n * On the elements coming before the target element, the function must return negative values;\n * on the elements coming after the target element, the function must return positive values.\n *\n * @return the index of the found element, if it is contained in the list within the specified range;\n * otherwise, the inverted insertion point `(-insertion point - 1)`.
 * The insertion point is defined as the index at which the element should be inserted,\n * so that the list (or the specified subrange of list) still remains sorted.\n * @sample
samples.collections.Collections.Lists.binarySearchWithComparisonFunction\n */\npublic fun \(<T>
List<T>.binarySearch(fromIndex: Int = 0, toIndex: Int = size, comparison: (T) -> Int): Int {
    rangeCheck(size, fromIndex, toIndex)

    var low = fromIndex
    var high = toIndex - 1

    while (low <= high) {
        val mid = (low + high).ushr(1) // safe from overflows
        val midVal = get(mid)
        val cmp = comparison(midVal)

        if (cmp < 0)
            low = mid + 1
        else if (cmp > 0)
            high = mid - 1
        else
            return mid // key found
    }
    return -(low + 1)  // key not found
}

/**
 * Checks that `from` and `to` are in\n * the range of [0..size] and throws an appropriate exception, if they aren't.
 */
private fun rangeCheck(size: Int, fromIndex: Int, toIndex: Int) {
    when {
        fromIndex > toIndex -> throw IllegalArgumentException("fromIndex ($fromIndex) is greater than toIndex ($toIndex).")
        fromIndex < 0 -> throw IndexOutOfBoundsException("fromIndex ($fromIndex) is less than zero.")
        toIndex > size -> throw IndexOutOfBoundsException("toIndex ($toIndex) is greater than size ($size).")
    }
}

@PublishedApi
@SinceKotlin("1.3")
internal expect fun checkIndexOverflow(index: Int): Int

@PublishedApi
@SinceKotlin("1.3")
internal expect fun checkCountOverflow(count: Int): Int

@PublishedApi
@SinceKotlin("1.3")
internal fun throwIndexOverflow() { throw ArithmeticException("Index overflow has happened.") }

@PublishedApi
@SinceKotlin("1.3")
internal fun throwCountOverflow() { throw ArithmeticException("Count overflow has happened.") }

@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("MapsKt")

package kotlin.collections

import kotlin.contracts.*

private object EmptyMap : Map<Any?, Nothing>, Serializable {
    private const val serialVersionUID: Long = 82467182954688274

    override fun equals(other: Any?): Boolean = other is Map<*, *> && other.isEmpty()
    override fun hashCode(): Int = 0
    override fun toString(): String = "{}"

    override val size: Int get() = 0
    override fun isEmpty(): Boolean = true

    override fun containsKey(key: Any?): Boolean = false
    override fun containsValue(value: Nothing): Boolean = false

    override fun get(key: Any?): Nothing? = null

    override fun readResolve(): Any = EmptyMap
}

/**
 * Returns an empty read-only map of specified type.
 * The returned map is serializable (JVM).
 * @sample samples.collections.Maps.Instantiation.emptyReadOnlyMap
 */
public fun <K, V> emptyMap(): Map<K, V> = @Suppress("UNCHECKED_CAST") (EmptyMap as Map<K, V>)

/**
 * Returns a new read-only map with the specified contents, given as a list of pairs
 * where the first component is the key and the second is the value.
 * Entries of the map are iterated in the order they were specified.
 * The returned map is serializable (JVM).
 * @sample samples.collections.Maps.Instantiation.mapFromPairs
 * @sample samples.collections.Maps.Instantiation.emptyReadOnlyMap
 */
public fun <K, V> mapOf(vararg pairs: Pair<K, V>): Map<K, V> =
    LinkedHashMap<K, V>(mapCapacity(pairs.size)).apply { putAll(pairs) }

/**
 * Returns an empty new [HashMap].
 * @sample
 */
public fun <K, V> hashMapOf(): HashMap<K, V> = HashMap()
defaultValue() else this

/**
 * Checks if the map contains the given key.
 * Returns the value corresponding to the given [key], or `null` if such a key is not present in the map.
 */

/**
 * Returns the value corresponding to the given [key], or `null` if such a key is not present in the map.
 */

/**
 * Allows to use the index operator for storing values in a mutable map.
 */

/**
 * Returns `true` if the map contains the specified [key]. Returns `false` if the map contains no such key.
 */
public inline fun <K> Map<K, V>.containsKey(key: K): Boolean = containsKey(key)

/**
 * Returns `true` if the map contains the specified [value].
 */
public inline fun <K, V> Map<K, V>.containsValue(value: V): Boolean = containsValue(value)

/**
 * Removes the specified key and its corresponding value from this map.
 */
public inline fun <K, V> Map<K, V>.remove(key: K): V? = remove(key)

/**
 * Returns the key component of the map entry.
 */
public inline operator fun <K, V> Map.Entry<K, V>.component1(): K = key

/**
 * Returns the value component of the map entry.
 */
public inline operator fun <K, V> Map.Entry<K, V>.component2(): V = value

/**
 * Converts entry to [Pair] with key being first component and value being second.
 */
public inline fun <K, V> Map.Entry<K, V>.toPair(): Pair<K, V> = toPair()

/**
 * Returns the value for the given key, or throws an exception if there is no such key in the map.
 */
public fun <K, V> Map<K, V>.getValue(key: K): V = getValue(key)

/**
 * Returns the value for the given key. If the key is not found in the map, calls the [defaultValue] function,
 * puts its result into the map under the given key and returns it.
 */

/**
 * Converts to [Pair] with key being first component and value being second.
 */
public inline fun <K, V> Map.Entry<K, V>.toPair(): Pair<K, V> = toPair()

/**
 * Returns the value for the given key, or throws an exception if there is no such key in the map.
 */
public fun <K, V> Map<K, V>.getOrNull(key: K): V? = getOrNull(key)

/**
 * Returns the value for the given key, or throws a `NoSuchElementException` if the key is not found in the map.
 */
public fun <K, V> Map<K, V>.getOrElse(key: K, defaultValue: () -> V): V = getOrElse(key, defaultValue)

/**
 * Returns the value for the given key, or throws a `NoSuchElementException` if the key is not found in the map.
 */

/**
 * Converts to [Pair] with key being first component and value being second.
 */
public inline fun <K, V> Map.Entry<K, V>.toPair(): Pair<K, V> = toPair()

/**
 * Returns the value for the given key, or throws a `NoSuchElementException` if the key is not found in the map.
 */
public fun <K, V> Map<K, V>.getOrElseOrNull(key: K, defaultValue: () -> V): V? = getOrElseOrNull(key, defaultValue)

/**
 * Returns the value for the given key, or throws a `NoSuchElementException` if the key is not found in the map.
 */
public fun <K, V> Map<K, V>.getOrElseOrDefaultOrNull(key: K, defaultValue: () -> V): V? = getOrElseOrDefaultOrNull(key, defaultValue)
Open Source Used In security_automation identity-guest-cs 1361
order of the original map.

* @sample samples.collections.Maps.Filtering

    \n    return filterTo(LinkedHashMap<K, V>(), predicate)
}

* @sample samples.collections.Maps.Filtering

    \n    return filterNotToLinkedHashMap<K, V>(predicate)
}

* @sample samples.collections.Maps.Filtering

    \n    return filterNotTo(LinkedHashMap<K, V>(), predicate)
}

* @sample samples.collections.Maps.Filtering

public inline fun <K, V> Map<out K, V>.toMap(): Map<K, V> = when (size) {
    0 -> emptyMap()
    1 -> mapOf(this@toMap)
    else -> toMap(LinkedHashMap<K, V>(mapCapacity(size)))
}

* @sample samples.collections.Maps.Filtering

    \n    return toMap(destination)
}

* @sample samples.collections.Maps.Filtering

    if (this.isEmpty()) mapOf(pair) else LinkedHashMap(this).apply { put(pair.first, pair.second) }

* @sample samples.collections.Maps.Filtering

    if (this.isEmpty()) pairs.toMap() else LinkedHashMap(this).apply { putAll(pairs) }

* @sample samples.collections.Maps.Filtering

public operator fun <K, V> Map<out K, V>.plus(pairs: Sequence<Pair<K, V>>): Map<K, V> = toMap(LinkedHashMap<K, V>(mapCapacity(size)))

* @sample samples.collections.Maps.Filtering

public fun <K, V> Sequence<Pair<K, V>>.toMap(): Map<K, V> = toMap(LinkedHashMap<K, V>(mapCapacity(size)))

* @sample samples.collections.Maps.Filtering

    destination.apply { putAll(this@toMap) }

* @sample samples.collections.Maps.Filtering

@SinceKotlin("1.1")
public fun <K, V> Map<out K, V>.toMap(): Map<K, V> = when (size) {
    0 -> emptyMap()
    1 -> toSingletonMap()
    else -> toMap(LinkedHashMap<K, V>(mapCapacity(size)))
}

* @sample samples.collections.Maps.Filtering

@SinceKotlin("1.1")
    destination.apply { putAll(this@toMap) }

* @sample samples.collections.Maps.Filtering

val this: Array<out Pair<K, V>>.toMap(): Map<K, V> = when (size) {
    0 -> emptyMap()
    1 -> mapOf(this@toMap)
    else -> toMap(LinkedHashMap<K, V>(mapCapacity(size)))
}

* @sample samples.collections.Maps.Filtering

val this: Array<out Pair<K, V>>.toMap(destination: M): M =
    destination.apply { putAll(this@toMap) }

* @sample samples.collections.Maps.Filtering

val this: Sequence<Pair<K, V>>.toMap(): Map<K, V> = when (size) {
    0 -> emptyMap()
    1 -> toSingletonMap()
    else -> toMap(LinkedHashMap<K, V>(mapCapacity(size)))
}

* @sample samples.collections.Maps.Filtering

val this: Sequence<Pair<K, V>>.toMap(destination: M): M =
    destination.apply { putAll(this@toMap) }

* @sample samples.collections.Maps.Filtering

val this: LinkedHashMap(this).toMap(): Map<K, V> = when (size) {
    0 -> emptyMap()
    1 -> mapOf(this@toMap)
    else -> toMap(LinkedHashMap<K, V>(mapCapacity(size)))
}

* @sample samples.collections.Maps.Filtering

val this: LinkedHashMap(this).toMap(destination: M): M =
    destination.apply { putAll(this@toMap) }

* @sample samples.collections.Maps.Filtering

val this: LinkedHashMap(this).plus(pair: Pair<K, V>): Map<K, V> = when (size) {
    0 -> emptyMap()
    1 -> mapOf(pair)
    else -> toMap(LinkedHashMap<K, V>(mapCapacity(size)))
}

* @sample samples.collections.Maps.Filtering

val this: LinkedHashMap(this).plus(pairs: Iterable<Pair<K, V>>): Map<K, V> = when (size) {
    0 -> emptyMap()
    1 -> mapOf(pairs)
    else -> toMap(LinkedHashMap<K, V>(mapCapacity(size)))
}

* @sample samples.collections.Maps.Filtering

val this: LinkedHashMap(this).plus(pairs: Sequence<Pair<K, V>>): Map<K, V> = when (size) {
    0 -> emptyMap()
    1 -> mapOf(pairs)
    else -> toMap(LinkedHashMap<K, V>(mapCapacity(size)))
}

* @sample samples.collections.Maps.Filtering

val this: MutableList<in K, in V>.toMap(): Map<K, V> = when (size) {
    0 -> emptyMap()
    1 -> mapOf(this@toMap)
    else -> toMap(LinkedHashMap<K, V>(mapCapacity(size)))
}

* @sample samples.collections.Maps.Filtering

val this: MutableList<in K, in V>.toMap(destination: M): M =
    destination.apply { putAll(this@toMap) }

* @sample samples.collections.Maps.Filtering

val this: MutableList<in K, in V>.plus(pair: Pair<K, V>): Map<K, V> = when (size) {
    0 -> emptyMap()
    1 -> mapOf(pair)
    else -> toMap(LinkedHashMap<K, V>(mapCapacity(size)))
}

* @sample samples.collections.Maps.Filtering

val this: MutableList<in K, in V>.plus(pairs: Iterable<Pair<K, V>>): Map<K, V> = when (size) {
    0 -> emptyMap()
    1 -> mapOf(pairs)
    else -> toMap(LinkedHashMap<K, V>(mapCapacity(size)))
}

* @sample samples.collections.Maps.Filtering

val this: MutableList<in K, in V>.plus(pairs: Sequence<Pair<K, V>>): Map<K, V> = when (size) {
    0 -> emptyMap()
    1 -> mapOf(pairs)
    else -> toMap(LinkedHashMap<K, V>(mapCapacity(size)))
}
LinkedHashMap(this).apply { putAll(pairs) } /* Creates a new read-only map by replacing or adding entries to this map from a given array of key-value [pairs]. */ The returned map preserves the entry iteration order of the original map. Those [pairs] with unique keys are iterated in the end in the order of [pairs] array. 

    if (this.isEmpty()) pairs.toMap() else LinkedHashMap(this).apply { putAll(pairs) } /* Creates a new read-only map by replacing or adding entries to this map from a given sequence of key-value [pairs]. */ The returned map preserves the entry iteration order of the original map. Those entries of another [map] that are missing in this map are iterated in the end in the order of that [map]. 

    val LinkedHashSetMap(this).apply { putAll(pairs) }.optimizeReadOnlyMap() /* Creates a new read-only map by replacing or adding entries to this map from another [map]. */ The returned map preserves the entry iteration order of the original map. 

    this.toMutableMap().apply { minusAssign(key) } /* Returns a map containing all entries of the original map except the entry with the given [key]. */ The returned map preserves the entry iteration order of the original map. 

    this.toMutableMap().apply { minusAssign(keys) } /* Removes all entries the keys of which are contained in the given [keys] collection. */ The returned map preserves the entry iteration order of the original map. 

    this.toMutableMap().apply { minusAssign(keys) } /* Removes all entries the keys of which are contained in the given [keys] sequence. */ The returned map preserves the entry iteration order of the original map. 

operator fun <K, V> Map<out K, V>.minusAssign(key: K): Unit
    remove(key) /* Removes all entries the keys of which are contained in the given [keys] array from this mutable map. */ 

operator fun <K, V> Map<out K, V>.minusAssign(keys: Sequence<K>): Unit
    this.keys.removeAll(keys) /* Removes all entries the keys of which are contained in the given [keys] sequence from this mutable map. */ 

operator fun <K, V> Map<out K, V>.minusAssign(keys: Iterable<K>): Unit
    this.keys.removeAll(keys) /* Removes all entries the keys of which are contained in the given [keys] array from this mutable map. */ 

operator fun <K, V> Map<out K, V>.minusAssign(keys: Array<K>): Unit
    this.keys.removeAll(keys) /* Removes all entries from the
keys of which are contained in the given [keys] sequence from this mutable map.

```kotlin
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public inline operator fun <K, V> MutableMap<K, V>.minusAssign(keys: Sequence<K>) { null // do not expose for now
    this.keys.removeAll(keys) // @PublishedApi
    @internal fun <K, V> Map<K, V>.optimizeReadOnlyMap() = when (size) {
        0 -> emptyMap()
        1 -> toSingletonMapOrSelf()
    }
    override fun hashCode(): Int = 0
    override fun toString(): String = ""
    override fun containsAll(elements: Collection<Nothing>): Boolean = elements.isEmpty()
}

public inline fun <K, V> MutableMap<K, V>.minusAssign(keys: Sequence<K>) {
    this.keys.removeAll(keys)
}

// do not expose for now
@PublishedApi
internal fun <K, V> Map<K, V>.optimizeReadOnlyMap() = when (size) {
    0 -> emptyMap()
    1 -> toSingletonMapOrSelf()
    else -> this
}
```

Use of this source code is governed by the Apache 2.0 license that can be found in the LICENSE.txt file.

```kotlin
@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("SetsKt")
@file:OptIn(kotlin.experimental.ExperimentalTypeInference::class)

package kotlin.collections

import kotlin.contracts.*

internal object EmptySet : Set<Nothing>, Serializable {
    private const val serialVersionUID: Long = 34066037487020532

    override fun equals(other: Any?): Boolean = other is Set<*> && other.isEmpty()
    override fun hashCode(): Int = 0
    override fun toString(): String = ""
    override fun isEmpty(): Boolean = true
    override fun contains(element: Nothing): Boolean = false
    override fun containsAll(elements: Collection<Nothing>): Boolean = elements.isEmpty()
    override fun iterator(): Iterator<Nothing> = EmptyIterator
}

fun <T> setOf(vararg elements: T): Set<T> = if (elements.size > 0) elements.toSet() else emptySet()

fun <T> setOfNotNull(element: T?): Set<T> = if (element != null) setOf(element) else emptySet()

fun <T : Any> setOfNotNull(elements: Collection<T>): Set<T> = emptySet()

fun <T> linkedSetOf(vararg elements: T): LinkedHashSet<T> = elements.toCollection(LinkedHashSet(mapCapacity(elements.size)))

fun <T> linkedSetOf(vararg elements: T): LinkedHashSet<T> = linkedSetOf(elements)
```

Builds a new read-only [Set] by populating a [MutableSet] using the given [builderAction][n] and returning a read-
only set with the same elements.[n] The set passed as a receiver to the [builderAction] is valid only inside that
function.

Using it outside of the function produces an unspecified behavior.

Elements of the set are iterated in the order they were added in the [builderAction].

---

* `buildSet(BuilderInference builderAction: MutatableSet<E>.() -> Unit): Set<E>`

* `buildSet(capacity: Int, @BuilderInference builderAction: MutatableSet<E>.() -> Unit): Set<E>`

---

* `toString(): String`:
  - Parses the string as a signed Byte number and returns the result or null if the string is not a valid representation of a number.
  - Parses the string as a signed Short number and returns the result or null if the string is not a valid representation of a number.
  - Parses the string as an Int number and returns the result or null if the string is not a valid representation of a number.

---

* `toString(radix: Int): String`:
  - Parses the string as a signed Byte number and returns the result or null if the string is not a valid representation of a number.
  - Parses the string as a signed Short number and returns the result or null if the string is not a valid representation of a number.
  - Parses the string as an Int number and returns the result or null if the string is not a valid representation of a number.

---

This file is copyrighted by JetBrains s.r.o. and Kotlin Programming Language contributors.

Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

---

Open Source Used in security_automation identity-guest-cs 1365
Int.MAX_VALUE
} else
    return null
} else
    {n
        start = 0
        isNegative = false
        limit = -Int.MAX_VALUE
    } else
    {n
        start = 0
        isNegative = false
        limit = -Int.MAX_VALUE
    }

    val limitForMaxRadix = (-Int.MAX_VALUE) / 36n
    var limitBeforeMul = limitForMaxRadix
    val result = 0
    for (i in start until length)
        {n
            val digit = digitOf(this[i], radix)

            if (digit < 0) return null
            if (result < limitBeforeMul)
                {n
                    if (result < limitBeforeMul)
                        {n
                            result *= radix
                        } else
                            {n
                                return null
                            }
                    if (result + digit > limitBeforeMul)
                        return null
                    result -= digit
                } else
                    {n
                        return null
                    }
        } else
            {n
                result = result * radix
            } else
                {n
                    result *= radix
                }

    return if (isNegative) result else -result
}

/**
 * Parses the string as a [Long] number and returns the result
 * or `null` if the string is not a valid representation of a number.
 */
@SinceKotlin("1.1")
public fun String.toLongOrNull(): Long? = toLongOrNull(radix = 10)

/**
 * Parses the string as a [Long] number and returns the result
 * or `null` if the string is not a valid representation of a number.
 *
 * @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.
 */
@SinceKotlin("1.1")
public fun String.toLongOrNull(radix: Int): Long? {

    val limit = Long.MAX_VALUE
    val limitBeforeMul = limitForMaxRadix
    val length = this.length
    if (length == 0) return null
    val start:
    val isNegative: Boolean
    val firstChar = this[0]
    if (firstChar < '0') {  // Possible leading sign
        if (length == 1) return null  // non-digit (possible sign) only, no digits after 'n' start = 1
        if (firstChar == '-')
            {n
                isNegative = true
                limit = Long.MIN_VALUE
            } else if (firstChar == '+')
                isNegative = false
            else
                return null

        else
            {n
                start = 0
                isNegative = false
                limit = -Long.MAX_VALUE
            } else
                {n
                    return null
                }
    } else
        {n
            start = 0
            isNegative = false
            limit = -Long.MAX_VALUE
        }

    val limitForMaxRadix = (-Long.MAX_VALUE) / 36n
    var limitBeforeMul = limitForMaxRadix
    var result = 0L
    for (i in start until length) {
        val digit = digitOf(this[i], radix)

        if (digit < 0) return null
        if (result < limitBeforeMul) {
            if (limitBeforeMul == limitForMaxRadix) {
                limitBeforeMul = limit / radix
            } else {
                limitBeforeMul = limit / radix
            }
        }

        result *= radix

        if (result < limit + digit) return null

        result -= digit
    }

    return if (isNegative) result else -result
}

internal fun numberFormatError(input: String): Nothing = throw NumberFormatException("Invalid number format: '$input'")

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */
@file:kotlin.jvm.JvmName("UnsignedKt")
@file:OptIn(ExperimentalUnsignedTypes::class)
package kotlin

@PublishedApi
internal fun uintCompare(v1: Int, v2: Int): Int = (v1 xor Int.MIN_VALUE).compareTo(v2 xor Int.MIN_VALUE)
@PublishedApi

@PublishedApi
internal fun uintDivide(v1: UInt, v2: UInt): UInt = (v1.toLong() / v2.toLong()).toUInt()
@PublishedApi
internal fun uintRemainder(v1: UInt, v2: UInt): UInt = (v1.toLong() % v2.toLong()).toUInt()

// Division and remainder are based on Guava's UnsignedLongs implementation
// Copyright 2011 The Guava Authors
@PublishedApi
internal fun ulongDivide(v1: ULong, v2: ULong): ULong {
    val dividend = v1.toLong()
    val divisor = v2.toLong()
    if (divisor < 0) { // i.e., divisor >= 2^63
        return if (v1 < v2)
            ULong(0)
        else
            ULong(1)
    } else
        {n
            return if (isNegative)
                result - result
            else
                result
            result = result % divisor
        }

    return if (isNegative) result else -result
}

@PublishedApi
internal fun ulongRemainder(v1: ULong, v2: ULong): ULong {
    val dividend = v1.toLong()
    val divisor = v2.toLong()
    if (divisor < 0) { // i.e., divisor >= 2^63
        return if (v1 < v2)
            dividend
        else
            0
    } else
        {n
            return if (isNegative)
                dividend - dividend
            else
                dividend
            dividend = dividend % divisor
        }

    return if (isNegative) result else -result
}

@PublishedApi
internal fun doubleToUInt(v: Double): UInt = when {
    v.isNaN() -> 0u
    v <= UInt.MIN_VALUE.toDouble() -> UInt.MIN_VALUE
    v >= 2^63.toDouble() -> UInt.MAX_VALUE
    else
        return UInt.valueOf((v - 2^63.toDouble()).toBigDecimal().toBigInteger().toDouble())
}
UInt.MAX_VALUE.toDouble() -> UInt.MAX_VALUE
    v <= Int.MAX_VALUE -> v.toInt().toUInt()
    else -
    (v - Int.MAX_VALUE).toInt().toUInt() + Int.MAX_VALUE.toUInt()      // Int.MAX_VALUE < v <
UInt.MAX_VALUE
}

@PublishedApi
internal fun doubleToULong(v: Double): ULong = when {
    v.isNaN() -> 0u
    v <= ULong.MIN_VALUE.toDouble() -> ULong.MIN_VALUE
    v >=
    ULong.MAX_VALUE.toDouble() -> ULong.MAX_VALUE
    v < Long.MAX_VALUE -
    v.toLong().toULong()    // Real values from Long.MAX_VALUE to (Long.MAX_VALUE + 1) are not
    representable in Double, so don't handle them.
    else -> (v - 9223372036854775808.0).toLong().toULong() + 9223372036854775808L      // Long.MAX_VALUE + 1 < v <
ULong.MAX_VALUE
}

@PublishedApi
internal fun uintToDouble(v: Int): Double = (v and
    Int.MAX_VALUE).toDouble() + (v ushr 31 shl 30).toDouble() * 2

@PublishedApi
internal fun
ulongToDouble(v: Long): Double = (v ushr 11).toDouble() * 2048 + (v and 2047)

internal fun
ulongToString(v: Long): String = ulongToString(v, 10)

internal fun
ulongToString(v: Long, base: Int): String {
    if (v >= 0) return v.toString(base)

    var quotient = ((v ushr 1) / base) shl 1
    var rem = v - quotient * base
    if (rem > base) {
        rem -= base
        quotient += 1
    }
    return quotient.toString(base) +
    rem.toString(base)
}

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language
 * contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the
 * license/LICENSE.txt file.
 */
@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("CollectionsKt")

package
kotlin.collections

/**
 * Given an [iterator] function constructs an [Iterable] instance that returns values through the
 * [Iterator] provided by that function.
 */
@PublishedApi
internal fun <T> Iterable(crossinline iterator: () -> Iterator<T>): Iterable<T> = object : Iterable<T> {
    override fun iterator(): Iterator<T> = iterator()
}

/**
 * A wrapper over another [Iterable] (or any other object that can produce an [Iterator]) that returns
 * an indexing iterator.
 */
internal class IndexingIterable<out T>(private val iteratorFactory: () -> Iterator<T>) : Iterable<IndexedValue<T>> {
    override
    fun iterator(): Iterator<IndexedValue<T>> = IndexingIterator(iteratorFactory())
}

/**
 * Returns the size of this iterable if it is known, or `null` otherwise.
 */
@PublishedApi
internal fun <T> Iterable<T>.collectionSizeOrNull(): Int? = if (this is Collection<*>) this.size else null

/**
 * Returns the size of this iterable if it is known, or the specified [default] value otherwise.
 */
@PublishedApi
internal fun <T> Iterable<T>.collectionSizeOrDefault(default: Int): Int = if (this is Collection<*>) this.size else default

/** Converts this collection to a set, when it's worth so and it doesn't change contains method behavior. */
internal fun <T> Iterable<T>.convertToSetForSetOperation(): Collection<T> =
    when (this) {
        is Set -> this
        is Collection ->
            when {
                source is Collection && source.size < 2 -> this
                else -> if (this.safeToConvertToSet()) toHashSet() else this
            }
        else -> toHashSet()
    }

/** Converts this collection to a set, when it's worth so and it doesn't change contains method behavior. */
internal fun <T> Iterable<T>.convertToSetForSetOperationWith(source: Iterable<T>): Collection<T> =
    when (this) {
        is Set -> this
        is Collection ->
            when {
                source is Collection && source.size < 2 -> this
                else -> if (this.safeToConvertToSet()) toHashSet() else this
            }
        else -> toHashSet()
    }

/*
 * Copyright 2010-2020
 * JetBrains s.r.o. and Kotlin Programming Language contributors.
 * * Use of this source code is governed by the Apache 2.0 license that can be found in the
 * license/LICENSE.txt file.
 */
Apache 2.0 license that can be found in the license/LICENSE.txt file.

An @file:kotlin.jvm.JvmName("SequencesKt")

import kotlin.random.Random

/package kotlin.sequences

import kotlin.random.Random

/**
 * Given an [Iterator] function constructs a [Sequence] that returns values through the [Iterator] provided by that function.
 * The values are evaluated lazily, and the sequence is potentially infinite.
 * @sample samples.collections.Sequences.Building.sequenceFromIterator
 */
@kotlin.internal.InlineOnly
public inline fun <T> Sequence(crossinline iterator: () -> Iterator<T>): Sequence<T> = object : Sequence<T> {
    override fun iterator(): Iterator<T> = iterator()
}

/**
 * Creates a sequence that returns all elements from this iterator. The sequence is constrained to be iterated only once.
 * @sample samples.collections.Sequences.Building.sequenceFromIterator
 */
public fun <T> Iterator<T>.asSequence(): Sequence<T> = Sequence { this }.constrainOnce()

/**
 * Creates a sequence that returns the specified values.
 * @sample samples.collections.Sequences.Building.sequenceOfValues
 */
public fun <T> sequenceOf(vararg elements: T): Sequence<T> = if (elements.isEmpty()) emptySequence() else elements.asSequence().onEmpty * Returns an empty sequence.

/**
 * Returns an empty sequence.
 */
public fun <T> emptySequence(): Sequence<T> = EmptySequence

private object EmptySequence : Sequence<Nothing>, DropTakeSequence<Nothing> {
    override fun iterator(): Iterator<Nothing> = EmptyIterator
    override fun drop(n: Int) = EmptySequence
    override fun take(n: Int) = EmptySequence
}

/**
 * Returns a sequence that iterates through the elements either of this sequence or, if this sequence turns out to be empty, of the sequence returned by [defaultValue] function.
 * @sample samples.collections.Sequences.Usage.sequenceOrEmpty
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun <T> Sequence<T>?.orEmpty(): Sequence<T> = this ?: emptySequence()

/**
 * Returns a sequence that iterates through the elements either of this sequence or, if this sequence turns out to be empty, of the sequence returned by [defaultValue] function.
 * @sample samples.collections.Sequences.Usage.sequenceIfEmpty
 */
@SinceKotlin("1.3")
public fun <T> Sequence<T>.ifEmpty(defaultValue: () -> Sequence<T>): Sequence<T> = sequence {
    val iterator = this@ifEmpty.iterator()
    if (iterator.hasNext()) {
        yieldAll(iterator)
    } else {
        yieldAll(defaultValue())
    }
}

/**
 * Returns a sequence of all elements from all sequences in this sequence.
 * The operation is _intermediate_ and _stateless_.
 * @sample samples.collections.Sequences.Transformations.flattenSequenceOfSequences
 */
public fun <T> Sequence<Sequence<T>>.flatten(): Sequence<T> = flatten { it.iterator() }

/**
 * Returns a sequence of all elements from all iterables in this sequence.
 * The operation is _intermediate_ and _stateless_.
 * @sample samples.collections.Sequences.Transformations.flattenSequenceOfIterable
 */
@kotlin.jvm.JvmName("flattenSequenceOfIterable")
public fun <T> Sequence<Iterable<T>>.flatten(): Sequence<T> = flatten { it.iterator() }

private fun <T, R> Sequence<T>.flatten(iterator: (T) -> Iterator<R>): Sequence<R> {
    if (this is TransformingSequence<*, *>) {
        return (this as TransformingSequence<*, T>).flatten(iterator)
    }
    return FlatteningSequence(this, { it }, iterator)
}

/**
 * Returns a pair of lists, where
 * *first* list is built from the first values of each pair from this sequence,
 * *second* list is built from the second values of each pair from this sequence.
 * The operation is _terminal_.
 * @sample samples.collections.Sequences.Transformations.unzip
 */
public fun <T, R> Sequence<Pair<T, R>>.unzip(): Pair<List<T>, List<R>> {
    val listT = ArrayList<T>()
    val listR = ArrayList<R>()
    for (pair in this) {
        listT.add(pair.first)
        listR.add(pair.second)
    }
    return listT to listR
}

/**
 * Returns a sequence that yields elements of this sequence randomly shuffled.
 * Note that every iteration of the sequence returns elements in a different order.
 * @sample samples.collections.Sequences.Transformations.shuffledSequence
 */
@SinceKotlin("1.4")
public fun <T> Sequence<T>.shuffled(): Sequence<T> = shuffled(Random)

/**
 * Returns a sequence that yields elements of this sequence randomly shuffled
 * using the specified [random] instance as the source of randomness.
 * Note that every iteration of the sequence returns elements in a different order.
 * @sample samples.collections.Sequences.Transformations.shuffledSequence
 */
@SinceKotlin("1.4")
public fun <T> Sequence<T>.shuffled(random: Random): Sequence<T> = sequence<T> {
    val buffer = toMutableList()
    while (buffer.isNotEmpty()) {
        val j = random.nextInt(buffer.size)
        val last = buffer.removeLast()
        val value = if (j < buffer.size) buffer.set(j, last)
    }
}
while (itemIterator == null) {
    if (!iterator.hasNext()) {
        return false
    } else {
        ensureItemIterator(): Boolean {
            if (itemIterator?.hasNext() == false)
                itemIterator = null

        override fun hasNext(): Boolean {
            return ensureItemIterator()
        }

        private fun
        if (!ensureItemIterator())
                throw NoSuchElementException()
            return itemIterator!!.next()

    iterator: (R) -> Iterator<E>
) : Sequence<E> {
    override fun iterator(): Iterator<E> = object : Iterator<E> {
        var iterator1 = sequence1.iterator()
        var iterator2 = sequence2.iterator()
        override fun next(): V {
            return transform(iterator1.next(), iterator2.next())
        }

        override fun hasNext(): Boolean {
            return iterator1.hasNext() && iterator2.hasNext()
        }
    }

    internal class FlatteningSequence<T, R, E>(sequence: Sequence<T>, transformer: (T) -> R, iterator: (R) -> Iterator<E>) : Sequence<E> {
        override fun iterator(): Iterator<E> = object : Iterator<E> {
            val iterator = sequence.iterator()
            var itemIterator: Iterator<T> = object : Iterator<T> {
                val iterator1 = sequence1.iterator()
                val iterator2 = sequence2.iterator()
                private val transform: (T) -> R
            } : Sequence<T> {
                override fun iterator(): Iterator<T> = object : Iterator<T> {
                    var itemIterator: Iterator<T> = object : Iterator<T> {
                        override fun hasNext(): Boolean {
                            return iterator1.hasNext() || iterator2.hasNext()
                        }

                        override fun next(): T {
                            if (iterator1.hasNext()) {
                                return iterator1.next()
                            } else {
                                return iterator2.next()
                            }
                        }
                    }
                }
            }
        }

        // Additional code...

        internal class MergingSequence<T1, T2, V>(sequence1: Sequence<T1>, sequence2: Sequence<T2>, transform: (T1, T2) -> V) : Sequence<V> {
            override fun iterator(): Iterator<V> = object : Iterator<V> {
                // Further implementation...
            }
        }

        /**
         * A sequence which combines values from the underlying [sequence] with their indices and returns them as [IndexedValue] objects. A sequence which returns the results of applying the given [transformer] function to the values in the underlying [sequence] along with the value itself.
         */

        internal class TransformingIndexedSequence<T, R>(sequence: Sequence<T>, transformer: (Int, T) -> R) : Sequence<R> {
            override fun iterator(): Iterator<R> = object : Iterator<R> {
                val iterator = sequence.iterator()
                var index = 0
                override fun next(): R {
                    val transformer = transformer(checkIndexOverflow(index++), iterator.next())
                    return transformer
                }

                override fun hasNext(): Boolean {
                    return iterator.hasNext()
                }
            }
        }

        /**
         * A sequence which returns the results of applying the given [transformer] function to the values in the underlying [sequence], where the transformer function takes the index of the value in the underlying * sequence along with the value itself. A sequence which combines values from the underlying [sequence] with their indices and returns them as [IndexedValue] objects.
         */

        internal class TransformingSequence<T, R>(sequence: Sequence<T>, transformer: (T) -> R) : Sequence<R> {
            override fun iterator(): Iterator<R> = object : Iterator<R> {
                val iterator = sequence.iterator()
                var nextState: Int = -1 // -1 for unknown, 0 for done, 1 for continue
                override fun hasNext(): Boolean = true,
                private val predicate: (T) -> Boolean
            } : Sequence<T> {
                override fun iterator(): Iterator<T> = object : Iterator<T> {
                    var nextState = 0
                    override fun hasNext(): Boolean = true
                    private val predicate: (T) -> Boolean
                } : Sequence<T> {
                    override fun override fun hasNext(): Boolean {
                        return hasNext()
                    }
                }
            }
        }

        /**
         * A sequence which returns the values from the underlying [sequence] that either match or do not match the specified [predicate]. A sequence that returns the values from the underlying [sequence] that either match or do not match the specified [predicate].
         */

        class FilteringSequence<T>(sequence: Sequence<T>, sendWhen: Boolean = true,
            private val predicate: (T) -> Boolean
        ) : Sequence<T> {
            override fun iterator(): Iterator<T> = object : Iterator<T> {
                var nextState: Int = -1 // -1 for unknown, 0 for done, 1 for continue
                override fun hasNext(): Boolean = true,
                private val predicate: (T) -> Boolean
            } : Sequence<T> {
                override fun iterator(): Iterator<T> = object : Iterator<T> {
                    var nextState = 0
                    override fun hasNext(): Boolean = true
                    private val predicate: (T) -> Boolean
                }
            }
        }

        /**
         * A sequence which returns the values the underlying [sequence] that either match or do not match the specified [predicate]. A sequence that returns the values from the underlying [sequence] that either match or do not match the specified [predicate].
         */

        class IndexingSequence<T>(sequence: Sequence<T>) : Sequence<IndexedValue<T>> {
            override fun iterator(): Iterator<IndexedValue<T>> = object : Iterator<IndexedValue<T>> {
                val iterator = sequence.iterator()
                var index = 0
                override fun next(): IndexedValue<T> {
                    // Additional implementation...
                }
            }
        }

        /**
         * A sequence which returns the values from the underlying [sequence] that either match or do not match the specified [predicate]. A sequence that returns the values from the underlying [sequence] that either match or do not match the specified [predicate].
         */

        class IndexedValueSequence<T>(sequence: Sequence<T>, transformer: (T) -> R) : Sequence<R> {
            override fun iterator(): Iterator<R> = object : Iterator<R> {
                // Additional implementation...
            }
        }

        /**
         * A sequence which combines values from the underlying
         */

        class TransformingSequence<T, R>
    }
}

/**
 * A sequence which returns the values from the underlying [sequence] that either match or do not match the specified [predicate]. A sequence that returns the values from the underlying [sequence] that either match or do not match the specified [predicate].
 */

internal class FilteringSequence<T>(sequence: Sequence<T>, sendWhen: Boolean = true,
    private val predicate: (T) -> Boolean
) : Sequence<T> {
    override fun iterator(): Iterator<T> = object : Iterator<T> {
        var nextState: Int = -1 // -1 for unknown, 0 for done, 1 for continue
        override fun hasNext(): Boolean = true,
        private val predicate: (T) -> Boolean
    } : Sequence<T> {
        override fun iterator(): Iterator<T> = object : Iterator<T> {
            var nextState = 0
            override fun hasNext(): Boolean = true
            private val predicate: (T) -> Boolean
        }
    }
}

/**
 * A sequence which combines values from the underlying
 */
override fun take(n: Int): Sequence<T> = (count + n).let {
    if (n1 < 0) TakeSequence(this, n) else
    SubSequence(sequence, count, n1)
}

override fun iterator(): Iterator<T> = object : Iterator<T> {
    val iterator = sequence.iterator()
    var left = count

    override fun next(): T {
        return iterator.next()
    }

    override fun hasNext(): Boolean {
        return iterator.hasNext()
    }
}

/**
 * A sequence that skips the values from the underlying [sequence] while the given
 * [predicate] returns `true` and returns all values after that.
 */
internal class DropWhileSequence<T>(
    private val sequence: Sequence<T>,
    private val predicate: (T) -> Boolean
) : Sequence<T> {
    override fun iterator(): Iterator<T> = object : Iterator<T> {
        val iterator = sequence.iterator()
        var dropState: Int = -1 // -1 for not dropping, 0 for normal iteration

        private fun drop() {
            while (iterator.hasNext()) {
                iterator.next()
                dropState--
            }
        }

        override fun next(): T {
            drop()
            return iterator.next()
        }

        override fun hasNext(): Boolean {
            drop()
            return iterator.hasNext()
        }
    }
}

internal class DistinctSequence<T, K>(
    private val source: Sequence<T>,
    private val keySelector: (T) -> K
) : Sequence<T> {
    override fun iterator(): Iterator<T> = DistinctIterator(source.iterator(), keySelector)
}

private class DistinctIterator<T, K>(
    private val source: Iterator<T>,
    private val keySelector: (T) -> K
) : AbstractIterator<T>() {
    private val observed = HashSet<K>()

    override fun computeNext() {
        while (source.hasNext()) {
            val next = source.next()
            val key = keySelector(next)
            if (observed.add(key)) {
                setNext(next)
                return
            }
        }

        done()
    }
}

/**
 * Returns a wrapper sequence that provides
 * values of this sequence, but ensures it can be iterated only one time.
 *
 * The operation is _intermediate_ and
 * _stateless_.
 *
 * @throws [IllegalStateException] is thrown on iterating the returned sequence from the second time.
 */
public fun <T> Sequence<T>.constrainOnce(): Sequence<T> {
    // as? does not work in js
    return if (this is ConstrainedOnceSequence<T>) this else ConstrainedOnceSequence(this)
}

/**
 * Returns a sequence which
 * invokes the function to calculate the next value based on
 * the previous one on each iteration.
 * If
 * [seed] is `null`, an empty sequence is produced.
 */
public fun <T : Any> generateSequence(nextFunction: () -> T?): Sequence<T> {
    return GeneratorSequence(nextFunction).constrainOnce()
}

open class GeneratorSequence<T : Any>(
    private val getInitialValue: () -> T?,
    private val getNextValue: (T) -> T?
) : Sequence<T> {
    override fun iterator(): Iterator<T> = object : Iterator<T> {
        var nextItem: T? = null
        var nextState: Int = -2 // -2 for initial unknown, -1 for next unknown, 0 for done, 1 for continue

        private fun calcNext() {
            nextItem = if (nextState == -2) getInitialValue() else getNextValue(nextItem!!)
            nextState = if (nextState == -2) 0 else if (nextState == 0) 1

            try {
                val result = calcNext()
                if (result == null) throw NoSuchElementException()
                if (result != null) return result
            } catch (e: NoSuchElementException) {
                if (nextState == 0) throw e
            }

            nextState = nextState + 1
        }

        override fun next(): T {
            if (nextState == 0)
                throw NoSuchElementException()

            return nextItem as T
        }

        override fun hasNext(): Boolean {
            return nextState > 0
        }
    }
}
Open Source Used In security_automation identity-guest-cs 1372

@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("PreconditionsKt")

package

import kotlin.contracts.contract

/**
 * Throws an [IllegalArgumentException] if the [value] is false.
 */
public fun require(value: Boolean): Unit {
    contract {
        returns() implies value
    }
    require(value) { "Failed requirement." }
}

/**
 * Throws an [IllegalArgumentException] with the result of calling 
 * [lazyMessage] if the [value] is false. Otherwise returns the not null value.
 */
public fun require(value: Boolean, lazyMessage: () -> Any): Unit {
    contract {
        returns() implies value
    }
    if (!value) {
        val message = lazyMessage()
        throw IllegalArgumentException(message.toString())
    } else {
        return value
    }
}

/**
 * Throws an [IllegalStateException] if the [value] is false.
 */
public fun check(value: Boolean): Unit {
    contract {
        returns() implies value
    }
    check(value) { "Check failed." }
}

/**
 * Throws an [IllegalStateException] with the result of calling 
 * [lazyMessage] if the [value] is false. Otherwise returns the not null value.
 */
public fun check(value: Boolean, lazyMessage: () -> Any): Unit {
    contract {
        returns() implies value
    }
    if (!value) {
        val message = lazyMessage()
        throw IllegalStateException(message.toString())
    } else {
        return value
    }
}

/**
 * Throws an [IllegalStateException] with the given [message].
 */
public fun error(message: Any): Nothing = throw
IllegalStateException(message.toString())

/*
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming
Language contributors. * Use of this source code is governed by the Apache 2.0 license that can be found in the
license/LICENSE.txt file. */

import kotlin.collections

/**
 * Returns an element at the given [index] or throws an
[IndexOutOfBoundsException] if the [index] is out of bounds of this array.

 * @sample samples.collections.Collections.Elements.elementAt
 */
public actual fun <T> Array<T>.elementAt(index: Int): T {
    return elementAtOrElse(index) { throw IndexOutOfBoundsException("index: $index, size: $size") }
}

/**
 * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.

 * @sample samples.collections.Collections.Elements.elementAt
 */
public actual fun String.forEach(elementAt: Int) {
    return forEachOrElse { throw IndexOutOfBoundsException("index: $index, size: $size") }
}

/**
 * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.

 * @sample samples.collections.Collections.Elements.elementAt
 */
public actual fun CharArray.elementAt(index: Int): Char {
    return elementAtOrElse(index) { throw IndexOutOfBoundsException("index: $index, size: $size") }
}

/**
 * Returns a [List] that wraps the original array.

 * @sample List<T>.asList()
 */
public actual fun <T> Array<T>.asList(): List<T> {
    return asListOrElse { throw IndexOutOfBoundsException("index: $index, size: $size") }
}

/**
 * Returns a [List] that wraps the original array.

 * @sample List<T>.asList()
 */
public actual fun CharArray.asList(): List<Char> {
    return asListOrElse { throw IndexOutOfBoundsException("index: $index, size: $size") }
}

/**
 * Returns a [List] that wraps the original array.

 * @sample List<T>.asList()
 */
public actual fun ByteArray.asList(): List<Byte> {
    return asListOrElse { throw IndexOutOfBoundsException("index: $index, size: $size") }
}

/**
 * Returns a [List] that wraps the original array.

 * @sample List<T>.asList()
 */
public actual fun IntArray.asList(): List<Int> {
    return asListOrElse { throw IndexOutOfBoundsException("index: $index, size: $size") }
}

/**
 * Returns a [List] that wraps the original array.

 * @sample List<T>.asList()
 */
public actual fun ShortArray.asList(): List<Short> {
    return asListOrElse { throw IndexOutOfBoundsException("index: $index, size: $size") }
}

/**
 * Returns a [List] that wraps the original array.

 * @sample List<T>.asList()
 */
public actual fun LongArray.asList(): List<Long> {
    return asListOrElse { throw IndexOutOfBoundsException("index: $index, size: $size") }
}

/**
 * Returns a [List] that wraps the original array.

 * @sample List<T>.asList()
 */
public actual fun FloatArray.asList(): List<Float> {
    return asListOrElse { throw IndexOutOfBoundsException("index: $index, size: $size") }
}

/**
 * Returns a [List] that wraps the original array.

 * @sample List<T>.asList()
 */
public actual fun DoubleArray.asList(): List<Double> {
    return asListOrElse { throw IndexOutOfBoundsException("index: $index, size: $size") }
}

/**
 * Returns a [List] that wraps the original array.

 * @sample List<T>.asList()
 */
public actual fun BooleanArray.asList(): List<Boolean> {
    return asListOrElse { throw IndexOutOfBoundsException("index: $index, size: $size") }
}

/**
 * Returns a [List] that wraps the original array.

 * @sample List<T>.asList()
 */
public actual fun CharArray.asList(): List<Char> {
    return asListOrElse { throw IndexOutOfBoundsException("index: $index, size: $size") }
}
this.unsafeCast<Array<Float>>()
    .asList()  

    @kotlin.internal.InlineOnly
    @kotlin.internal.InlineOnly
    public actual inline fun DoubleArray.asList(): List<Double> {
        return
            this.unsafeCast<Array<Double>>()
                .asList()  

    @kotlin.internal.InlineOnly
    @kotlin.internal.InlineOnly
    public actual inline fun BooleanArray.asList(): List<Boolean> {
        return
            this.unsafeCast<Array<Boolean>>()
                .asList()  

    public actual fun CharArray.asList(): List<Char> {
        return object : AbstractList<Char>(), RandomAccess {

            override val size: Int get() = this@asList.size
            override fun isEmpty(): Boolean = this@asList.isEmpty()
            override fun contains(element: Char): Boolean = this@asList.contains(element)
            override fun get(index: Int): Char {
                AbstractList.checkElementIndex(index, size)
                return this@asList[index]
            }
            override fun indexOf(element: Char): Int {
                @Suppress("USELESS_CAST")
                if ((element as Any?) !is Char) return -1
                return this@asList.indexOf(element)
            }
            override fun lastIndexOf(element: Char): Int {
                @Suppress("USELESS_CAST")
                if ((element as Any?) !is Char) return -1
                return this@asList.lastIndexOf(element)
            }
        }
    }

    /**
     * Returns a 
     * List that wraps the original array.
     * 
     * The elements of other types are compared for equality with the 
     * equals<Any.equals> function.
     * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
     */
    @SinceKotlin("1.1")
    @library("arrayDeepHashCode")
    public actual infix fun <T> Array<out T>?.contentDeepHashCode(): Int {
        return this.contentDeepHashCode()
    }

    /**
     * Returns a string representation of the contents of this array as if it is a 
     * List.
     * Nested arrays are treated as lists too.
     * If any of arrays contains itself on any nesting level the behavior is undefined.
     * The elements are compared for equality with the 
     * equals<Any.equals> function.
     * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
     */
    @SinceKotlin("1.1")
    @library("arrayDeepToString")
    public actual infix fun <T> Array<out T>?.contentDeepToString(): String {
        return this.contentDeepToString()
    }

    /**
     * Returns `true` if the two specified arrays are *structurally* equal to one another,
     * i.e. contain the same number of the same elements in the same order.
     * Nested arrays are treated as lists too.
     * If any of arrays contains itself on any nesting level the behavior is undefined.
     * The elements are compared for equality with the 
     * equals<Any.equals> function.
     * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
     */
    @SinceKotlin("1.1")
    @library("arrayDeepToString")
    public actual infix fun <T> Array<out T>?.contentDeepToString(): String {
        return this.contentDeepToString()
    }

    /**
     * Returns `false` if the two specified arrays are *structurally* equal to one another,
     * i.e. contain the same number of the same elements in the same order.
     * Nested arrays are treated as lists too.
     * If any of arrays contains itself on any nesting level the behavior is undefined.
     */
    @SinceKotlin("1.1")
    @library("arrayDeepEquals")
    public actual infix fun <T> Array<out T>?.contentDeepEquals(other: Array<out T>): Boolean {
        return this.contentDeepEquals(other)
    }

    /**
     * Returns `true` if the two specified arrays are *deeply* equal to one another,
     * i.e. contain the same number of the same elements in the same order.
     * Nested arrays are treated as lists too.
     * If any of arrays contains itself on any nesting level the behavior is undefined.
     * The elements of other types are compared for equality with the 
     * equals<Any.equals> function.
     * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
     */
    @SinceKotlin("1.1")
    @library("arrayDeepHashCode")
    public actual infix fun <T> Array<out T>?.contentDeepHashCode(): Int {
        return this.contentDeepHashCode()
    }

    /**
     * Returns a hash code based on the contents of this array as if it is a 
     * List.
     * Nested arrays are treated as lists too.
     * If any of arrays contains itself on any nesting level the behavior is undefined.
     * The elements are compared for equality with the 
     * equals<Any.equals> function.
     * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
     */
    @SinceKotlin("1.1")
    @library("arrayDeepToString")
    public actual infix fun <T> Array<out T>?.contentDeepToString(): String {
        return this.contentDeepToString()
    }

    /**
     * Returns a hash code based on the contents of this array as if it is a 
     * List.
     * Nested arrays are treated as lists too.
     * If any of arrays contains itself on any nesting level the behavior is undefined.
     * The elements are compared for equality with the 
     * equals<Any.equals> function.
     * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
     */
    @SinceKotlin("1.1")
    @library("arrayDeepEquals")
    public actual infix fun <T> Array<out T>?.contentDeepEquals(other: Array<out T>): Boolean {
        return this.contentDeepEquals(other)
    }

    /**
     * Returns a hash code based on the contents of this array as if it is a 
     * List.
     * Nested arrays are treated as lists too.
     * If any of arrays contains itself on any nesting level the behavior is undefined.
     * The elements are compared for equality with the 
     * equals<Any.equals> function.
     * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
     */
    @SinceKotlin("1.1")
    @library("arrayDeepHashCode")
    public actual infix fun <T> Array<out T>?.contentDeepHashCode(): Int {
        return this.contentDeepHashCode()
    }

    /**
     * Returns a string representation of the contents of this array as if it is a 
     * List.
     * Nested arrays are treated as lists too.
     * If any of arrays contains itself on any nesting level the behavior is undefined.
     * The elements are compared for equality with the 
     * equals<Any.equals> function.
     * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
     */
    @SinceKotlin("1.1")
    @library("arrayDeepToString")
    public actual infix fun <T> Array<out T>?.contentDeepToString(): String {
        return this.contentDeepToString()
    }

    /**
     * Returns a hash code based on the contents of this array as if it is a 
     * List.
     * Nested arrays are treated as lists too.
     * If any of arrays contains itself on any nesting level the behavior is undefined.
     * The elements are compared for equality with the 
     * equals<Any.equals> function.
     * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
     */
    @SinceKotlin("1.1")
    @library("arrayDeepEquals")
    public actual infix fun <T> Array<out T>?.contentDeepEquals(other: Array<out T>): Boolean {
        return this.contentDeepEquals(other)
    }

    /**
     * Returns a string representation of the contents of this array as if it is a 
     * List.
     * Nested arrays are treated as lists too.
     * If any of arrays contains itself on any nesting level the behavior is undefined.
     * The elements are compared for equality with the 
     * equals<Any.equals> function.
     * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
     */
    @SinceKotlin("1.1")
    @library("arrayDeepHashCode")
    public actual infix fun <T> Array<out T>?.contentDeepHashCode(): Int {
        return this.contentDeepHashCode()
    }

    /**
     * Returns a string representation of the contents of this array as if it is a 
     * List.
     * Nested arrays are treated as lists too.
     * If any of arrays contains itself on any nesting level the behavior is undefined.
     * The elements are compared for equality with the 
     * equals<Any.equals> function.
     * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
     */
    @SinceKotlin("1.1")
    @library("arrayDeepToString")
    public actual infix fun <T> Array<out T>?.contentDeepToString(): String {
        return this.contentDeepToString()
    }

    /**
     * Returns a hash code based on the contents of this array as if it is a 
     * List.
     * Nested arrays are treated as lists too.
     * If any of arrays contains itself on any nesting level the behavior is undefined.
     * The elements are compared for equality with the 
     * equals<Any.equals> function.
     * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
     */
    @SinceKotlin("1.1")
    @library("arrayDeepEquals")
    public actual infix fun <T> Array<out T>?.contentDeepEquals(other: Array<out T>): Boolean {
        return this.contentDeepEquals(other)
    }

    /**
     * Returns a string representation of the contents of this array as if it is a 
     * List.
     * Nested arrays are treated as lists too.
     * If any of arrays contains itself on any nesting level the behavior is undefined.
     * The elements are compared for equality with the 
     * equals<Any.equals> function.
     * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
     */
    @SinceKotlin("1.1")
    @library("arrayDeepHashCode")
    public actual infix fun <T> Array<out T>?.contentDeepHashCode(): Int {
        return this.contentDeepHashCode()
    }

    /**
     * Returns a string representation of the contents of this array as if it is a 
     * List.
     * Nested arrays are treated as lists too.
     * If any of arrays contains itself on any nesting level the behavior is undefined.
     * The elements are compared for equality with the 
     * equals<Any.equals> function.
     * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
     */
    @SinceKotlin("1.1")
    @library("arrayDeepToString")
    public actual infix fun <T> Array<out T>?.contentDeepToString(): String {
        return this.contentDeepToString()
    }
it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.

```kotlin
fun <T> Array<out T>.contentEquals(other: Array<out T>): Boolean {
    return this.contentEquals(other)
}
```

The elements are compared for equality with the `[equals][Any.equals]` function. For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.

```kotlin
ByteArray.contentEquals(other: ByteArray): Boolean {
    return this.contentEquals(other)
}
```

The elements are compared for equality with the `[equals][Any.equals]` function. For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.

```kotlin
ShortArray.contentEquals(other: ShortArray): Boolean {
    return this.contentEquals(other)
}
```

The elements are compared for equality with the `[equals][Any.equals]` function. For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.

```kotlin
IntArray.contentEquals(other: IntArray): Boolean {
    return this.contentEquals(other)
}
```

The elements are compared for equality with the `[equals][Any.equals]` function. For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.

```kotlin
LongArray.contentEquals(other: LongArray): Boolean {
    return this.contentEquals(other)
}
```

The elements are compared for equality with the `[equals][Any.equals]` function. For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.

```kotlin
FloatArray.contentEquals(other: FloatArray): Boolean {
    return this.contentEquals(other)
}
```

The elements are compared for equality with the `[equals][Any.equals]` function. For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.

```kotlin
DoubleArray.contentEquals(other: DoubleArray): Boolean {
    return this.contentEquals(other)
}
```

The elements are compared for equality with the `[equals][Any.equals]` function. For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.

```kotlin
BooleanArray.contentEquals(other: BooleanArray): Boolean {
    return this.contentEquals(other)
}
```

The elements are compared for equality with the `[equals][Any.equals]` function. For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.}

Open Source Used in security_automation identity-guest-cs 1375
```kotlin
/**
 * Returns `true` if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order. The elements are compared for equality with the [equals][Any.equals] function.
 * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
 */
@SinceKotlin("1.4")
@library("arrayEquals")
public actual infix fun ByteArray?.contentEquals(other: ByteArray?): Boolean {
    definedExternally
}

/**
 * Returns `true` if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order. The elements are compared for equality with the [equals][Any.equals] function.
 * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
 */
@SinceKotlin("1.4")
@library("arrayEquals")
public actual infix fun ShortArray?.contentEquals(other: ShortArray?): Boolean {
    definedExternally
}

/**
 * Returns `true` if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order. The elements are compared for equality with the [equals][Any.equals] function.
 * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
 */
@SinceKotlin("1.4")
@library("arrayEquals")
public actual infix fun IntArray?.contentEquals(other: IntArray?): Boolean {
    definedExternally
}

/**
 * Returns `true` if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order. The elements are compared for equality with the [equals][Any.equals] function.
 * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
 */
@SinceKotlin("1.4")
@library("arrayEquals")
public actual infix fun LongArray?.contentEquals(other: LongArray?): Boolean {
    definedExternally
}

/**
 * Returns `true` if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order. The elements are compared for equality with the [equals][Any.equals] function.
 * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
 */
@SinceKotlin("1.4")
@library("arrayEquals")
public actual infix fun FloatArray?.contentEquals(other: FloatArray?): Boolean {
    definedExternally
}

/**
 * Returns `true` if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order. The elements are compared for equality with the [equals][Any.equals] function.
 * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
 */
@SinceKotlin("1.4")
@library("arrayEquals")
public actual infix fun DoubleArray?.contentEquals(other: DoubleArray?): Boolean {
    definedExternally
}
```

---

Open Source Used In security_automation identity-guest-cs 1376
as if it is [List].

*\@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.")@SinceKotlin("1.1")@DeprecatedSinceKotlin(hidenSince = "1.4")\public actual fun <T> Array\<out T>\.contentHashCode(): Int {\n\t\nreturn this\.contentHashCode()\n}\n\n*\@DeprecatedSinceKotlin(hidenSince = "1.4")\public actual fun ByteArray\.contentHashCode(): Int {\n\t\nreturn this\.contentHashCode()\n}\n\n*\@DeprecatedSinceKotlin(hidenSince = "1.4")\public actual fun ShortArray\.contentHashCode(): Int {\n\t\nreturn this\.contentHashCode()\n}\n\n*\@DeprecatedSinceKotlin(hidenSince = "1.4")\public actual fun IntArray\.contentHashCode(): Int {\n\t\nreturn this\.contentHashCode()\n}\n\n*\@DeprecatedSinceKotlin(hidenSince = "1.4")\public actual fun LongArray\.contentHashCode(): Int {\n\t\nreturn this\.contentHashCode()\n}\n\n*\@DeprecatedSinceKotlin(hidenSince = "1.4")\public actual fun FloatArray\.contentHashCode(): Int {\n\t\nreturn this\.contentHashCode()\n}\n\n*\@DeprecatedSinceKotlin(hidenSince = "1.4")\public actual fun DoubleArray\.contentHashCode(): Int {\n\t\nreturn this\.contentHashCode()\n}\n\n*\@DeprecatedSinceKotlin(hidenSince = "1.4")\public actual fun BooleanArray\.contentHashCode(): Int {\n\t\nreturn this\.contentHashCode()\n}\n\n*\@DeprecatedSinceKotlin(hidenSince = "1.4")\public actual fun CharArray\.contentHashCode(): Int {\n\t\nreturn this\.contentHashCode()\n}\n\n*\@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.")@SinceKotlin("1.1")@DeprecatedSinceKotlin(hidenSince = "1.4")\public actual fun ByteArray\.contentHashCode(): Int {\n\t\nreturn this\.contentHashCode()\n}\n\n*\@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.")@SinceKotlin("1.1")@DeprecatedSinceKotlin(hidenSince = "1.4")\public actual fun ShortArray\.contentHashCode(): Int {\n\t\nreturn this\.contentHashCode()\n}\n\n*\@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.")@SinceKotlin("1.1")@DeprecatedSinceKotlin(hidenSince = "1.4")\public actual fun IntArray\.contentHashCode(): Int {\n\t\nreturn this\.contentHashCode()\n}\n\n*\@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.")@SinceKotlin("1.1")@DeprecatedSinceKotlin(hidenSince = "1.4")\public actual fun LongArray\.contentHashCode(): Int {\n\t\nreturn this\.contentHashCode()\n}\n\n*\@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.")@SinceKotlin("1.1")@DeprecatedSinceKotlin(hidenSince = "1.4")\public actual fun FloatArray\.contentHashCode(): Int {\n\t\nreturn this\.contentHashCode()\n}\n\n*\@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.")@SinceKotlin("1.1")@DeprecatedSinceKotlin(hidenSince = "1.4")\public actual fun DoubleArray\.contentHashCode(): Int {\n\t\nreturn this\.contentHashCode()\n}\n\n*\@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.")@SinceKotlin("1.1")@DeprecatedSinceKotlin(hidenSince = "1.4")\public actual fun BooleanArray\.contentHashCode(): Int {\n\t\nreturn this\.contentHashCode()\n}\n\n*\@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.")@SinceKotlin("1.1")@DeprecatedSinceKotlin(hidenSince = "1.4")\public actual fun CharArray\.contentHashCode(): Int {\n\t\nreturn this\.contentHashCode()\n}\n\n*\@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.")@SinceKotlin("1.1")@DeprecatedSinceKotlin(hidenSince = "1.4")\public actual fun ByteArray\.contentHashCode(): Int {\n\t\nreturn this\.contentHashCode()\n}\n\n*\@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.")@SinceKotlin("1.1")@DeprecatedSinceKotlin(hidenSince = "1.4")\public actual fun ShortArray\.contentHashCode(): Int {\n\t\nreturn this\.contentHashCode()\n}\n\n*\@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.")@SinceKotlin("1.1")@DeprecatedSinceKotlin(hidenSince = "1.4")\public actual fun IntArray\.contentHashCode(): Int {\n\t\nreturn this\.contentHashCode()\n}\n\n*\@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.")@SinceKotlin("1.1")@DeprecatedSinceKotlin(hidenSince = "1.4")\public actual fun LongArray\.contentHashCode(): Int {\n\t\nreturn this\.contentHashCode()\n}\n\n*\@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.")@SinceKotlin("1.1")@DeprecatedSinceKotlin(hidenSince = "1.4")\public actual fun FloatArray\.contentHashCode(): Int {\n\t\nreturn this\.contentHashCode()\n}\n\n*\@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.")@SinceKotlin("1.1")@DeprecatedSinceKotlin(hidenSince = "1.4")\public actual fun DoubleArray\.contentHashCode(): Int {\n\t\nreturn this\.contentHashCode()\n}\n\n*\@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.")@SinceKotlin("1.1")@DeprecatedSinceKotlin(hidenSince = "1.4")\public actual fun BooleanArray\.contentHashCode(): Int {\n\t\nreturn this\.contentHashCode()\n}\n\n*\@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.")@SinceKotlin("1.1")@DeprecatedSinceKotlin(hidenSince = "1.4")\public actual fun CharArray\.contentHashCode(): Int {\n\t\nreturn this\.contentHashCode()\n}\n\n*\@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.")@SinceKotlin("1.1")@DeprecatedSinceKotlin(hidenSince = "1.4")\public actual fun ByteArray\.contentHashCode(): Int {\n\t\nreturn this\.contentHashCode()\n}\n\n*\@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.")@SinceKotlin("1.1")@DeprecatedSinceKotlin(hidenSince = "1.4")\public actual fun ShortArray\.contentHashCode(): Int {\n\t\nreturn this\.contentHashCode()\n}\n\n*\@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.")@SinceKotlin("1.1")@DeprecatedSinceKotlin(hidenSince = "1.4")\public actual fun IntArray\.contentHashCode(): Int {\n\t\nreturn this\.contentHashCode()\n}\n\n*\@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.")@SinceKotlin("1.1")@DeprecatedSinceKotlin(hidenSince = "1.4")\public actual fun LongArray\.contentHashCode(): Int {\n\t\nreturn this\.contentHashCode()\n}\n\n*\@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.")@SinceKotlin("1.1")@DeprecatedSinceKotlin(hidenSince = "1.4")\public actual fun FloatArray\.contentHashCode(): Int {\n\t\nreturn this\.contentHashCode()\n}\n\n*\@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.")@SinceKotlin("1.1")@DeprecatedSinceKotlin(hidenSince = "1.4")\public actual function

samples.collections.Arrays.ContentOperations.contentToString
*\@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.")@SinceKotlin("1.1")@DeprecatedSinceKotlin(hidenSince = "1.4")\public actual fun
actual fun <T> Array<out T>.contentToString(): String {
    return this.contentToString()
}

/**
 * Returns a string representation of the contents of the specified array as if it is [List].
 * @sample samples.collections.Arrays.ContentOperations.contentToString
 */
@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.")
@SinceKotlin("1.1")
public actual fun ByteArray.contentToString(): String {
    return this.contentToString()
}

/**
 * Returns a string representation of the contents of the specified array as if it is [List].
 * @sample samples.collections.Arrays.ContentOperations.contentToString
 */
@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.")
@SinceKotlin("1.1")
public actual fun ShortArray.contentToString(): String {
    return this.contentToString()
}

/**
 * Returns a string representation of the contents of the specified array as if it is [List].
 * @sample samples.collections.Arrays.ContentOperations.contentToString
 */
@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.")
@SinceKotlin("1.1")
public actual fun IntArray.contentToString(): String {
    return this.contentToString()
}

/**
 * Returns a string representation of the contents of the specified array as if it is [List].
 * @sample samples.collections.Arrays.ContentOperations.contentToString
 */
@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.")
@SinceKotlin("1.1")
public actual fun LongArray.contentToString(): String {
    return this.contentToString()
}

/**
 * Returns a string representation of the contents of the specified array as if it is [List].
 * @sample samples.collections.Arrays.ContentOperations.contentToString
 */
@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.")
@SinceKotlin("1.1")
public actual fun FloatArray.contentToString(): String {
    return this.contentToString()
}

/**
 * Returns a string representation of the contents of the specified array as if it is [List].
 * @sample samples.collections.Arrays.ContentOperations.contentToString
 */
@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.")
@SinceKotlin("1.1")
public actual fun DoubleArray.contentToString(): String {
    return this.contentToString()
}

/**
 * Returns a string representation of the contents of the specified array as if it is [List].
 * @sample samples.collections.Arrays.ContentOperations.contentToString
 */
@SinceKotlin("1.4")
@library("arrayToString")
public actual fun <T> Array<out T>?.contentToString(): String {
    definedExternally
}

/**
 * Returns a string representation of the contents of the specified array as if it is [List].
 * @sample samples.collections.Arrays.ContentOperations.contentToString
 */
@SinceKotlin("1.4")
@library("arrayToString")
public actual fun ByteArray?.contentToString(): String {
    definedExternally
}

/**
 * Returns a string representation of the contents of the specified array as if it is [List].
 * @sample samples.collections.Arrays.ContentOperations.contentToString
 */
@SinceKotlin("1.4")
@library("arrayToString")
public actual fun ShortArray?.contentToString(): String {
    definedExternally
}

/**
 * Returns a string representation of the contents of the specified array as if it is [List].
 * @sample samples.collections.Arrays.ContentOperations.contentToString
 */
@SinceKotlin("1.4")
@library("arrayToString")
public actual fun IntArray?.contentToString(): String {
    definedExternally
}

/**
 * Returns a string representation of the contents of the specified array as if it is [List].
 * @sample samples.collections.Arrays.ContentOperations.contentToString
 */
@SinceKotlin("1.4")
@library("arrayToString")
public actual fun LongArray?.contentToString(): String
```kotlin
@SinceKotlin("1.4")
@library("arrayToString")
public actual fun FloatArray?.contentToString(): String {
    definedExternally
}

@SinceKotlin("1.4")
@library("arrayToString")
public actual fun DoubleArray?.contentToString(): String {
    definedExternally
}

@SinceKotlin("1.4")
@library("arrayToString")
public actual fun BooleanArray?.contentToString(): String {
    definedExternally
}

@SinceKotlin("1.4")
@library("arrayToString")
public actual fun CharArray?.contentToString(): String {
    definedExternally
}

@SinceKotlin("1.4")
@library("arrayToString")
public actual fun ByteArray.copyInto(destination: ByteArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): ByteArray {
    arrayCopy(this.unsafeCast<Array<Byte>>(), destination.unsafeCast<Array<Byte>>(), destinationOffset, startIndex, endIndex)
    return destination
}
```

Open Source Used In security_automation identity-guest-cs 1379
*\@SinceKotlin("1.3")*\@kotlin.internal.InlineOnly\n@Suppress("\ACTUAL_FUNCTION_WITH_DEFAULT_ArgUMENTS")\n\public actual inline fun ShortArray.copyInto(destination: ShortArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): ShortArray {\n    arrayCopy(this.unsafeCast<Array<Short>>(), destination.unsafeCast<Array<Short>>(), destinationOffset, startIndex, endIndex)\n    return destination\n\n* Copies this array or its subrange into the [destination] array and returns that array.\n* It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.\n* \n* @param destination the array to copy to.\n* @param destinationOffset the position in the [destination] array to copy to, 0 by default.\n* @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.\n* @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.\n* @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.\n* \n\n*\@SinceKotlin("1.3")*\@kotlin.internal.InlineOnly\n@Suppress("\ACTUAL_FUNCTION_WITH_DEFAULT_ArgUMENTS")\n\public actual inline fun IntArray.copyInto(destination: IntArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): IntArray {
    arrayCopy(this.unsafeCast<Array<Int>>(), destination.unsafeCast<Array<Int>>(), destinationOffset, start

```
IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`. In this case, this function throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset]. In this case, the array indices range.

It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range. When [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`, this function throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset]. When [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`, this function throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset]. It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range. When [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`, this function throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset]. When [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`, this function throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset]. In this case, the array indices range.
filled with zero values.\n * @sample samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf\n *\n
anpublic actual fun DoubleArray.copyOf(newSize: Int): DoubleArray {\n    require(newSize >= 0) { "Invalid new array size: $newSize." }\n    return fillFrom(this, DoubleArray(newSize))\n}\n\n/**\n * Returns new array which is a copy of the original array, resized to the given [newSize].\n * The copy is either truncated or padded at the end with 'false' values if necessary.\n * @sample samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf\n *\n
anactual fun BooleanArray.copyOf(newSize: Int): BooleanArray {\n    require(newSize >= 0) { "Invalid new array size: $newSize." }\n    return withType("BooleanArray", arrayCopyResize(this, newSize, false))\n}\n\n/**\n * Returns new array which is a copy of the original array, resized to the given [newSize].\n * The copy is either truncated or padded at the end with null char (\u0000) values if necessary.\n *\n
anactual fun CharArray.copyOf(newSize: Int): CharArray {\n    require(newSize >= 0) { "Invalid new array size: $newSize." }\n    return withType("CharArray", fillFrom(this, CharArray(newSize)))\n}\n\n/**\n * Returns new array which is a copy of the original array, resized to the given [newSize].\n * The copy is either truncated or padded at the end with null values if necessary.\n *\n
anactual fun IntArray.copyOf(newSize: Int): IntArray {\n    require(newSize >= 0) { "Invalid new array size: $newSize." }\n    return withType("IntArray", arrayCopyResize(this, newSize, null))\n}\n\n/**\n * Returns a new array which is a copy of the specified range of the original array.
 *\n * @param fromIndex the start of the range (inclusive) to copy.
 * @param toIndex the end of the range (exclusive) to copy.
 *\n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 *\n
anpublic actual fun <T> ByteArray.copyOfRange(fromIndex: Int, toIndex: Int): ByteArray {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n    return this.asDynamic().slice(fromIndex, toIndex)\n}\n\n/**\n * Returns a new array which is a copy of the specified range of the original array.
 *\n * @param fromIndex the start of the range (inclusive) to copy.
 * @param toIndex the end of the range (exclusive) to copy.
 *\n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 *\n
anpublic actual fun <T> ShortArray.copyOfRange(fromIndex: Int, toIndex: Int): ShortArray {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n    return this.asDynamic().slice(fromIndex, toIndex)\n}\n\n/**\n * Returns a new array which is a copy of the specified range of the original array.
 *\n * @param fromIndex the start of the range (inclusive) to copy.
 * @param toIndex the end of the range (exclusive) to copy.
 *\n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 *\n
anpublic actual fun IntArray.copyOfRange(fromIndex: Int, toIndex: Int): IntArray {\n
AbstractList.checkRangeIndexes(fromIndex, toIndex, size) return this.asDynamic().slice(fromIndex, toIndex)\n\n/**
 * Returns a new array which is a copy of the specified range of the original array.
 * @param fromIndex the start of the range (inclusive) to copy.
 * @param toIndex the end of the range (exclusive) to copy.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
public actual fun LongArray.copyOfRange(fromIndex: Int, toIndex: Int): LongArray {\n    this.asDynamic().slice(fromIndex, toIndex)\n\n/**
 * Returns a new array which is a copy of the specified range of the original array.
 * @param fromIndex the start of the range (inclusive) to copy.
 * @param toIndex the end of the range (exclusive) to copy.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
public actual fun FloatArray.copyOfRange(fromIndex: Int, toIndex: Int): FloatArray {\n    this.asDynamic().slice(fromIndex, toIndex)\n\n/**
 * Returns a new array which is a copy of the specified range of the original array.
 * @param fromIndex the start of the range (inclusive) to copy.
 * @param toIndex the end of the range (exclusive) to copy.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
public actual fun DoubleArray.copyOfRange(fromIndex: Int, toIndex: Int): DoubleArray {\n    this.asDynamic().slice(fromIndex, toIndex)\n\n/**
 * Returns a new array which is a copy of the specified range of the original array.
 * @param fromIndex the start of the range (inclusive) to copy.
 * @param toIndex the end of the range (exclusive) to copy.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
public actual fun BooleanArray.copyOfRange(fromIndex: Int, toIndex: Int): BooleanArray {\n    this.asDynamic().slice(fromIndex, toIndex)\n\n/**
 * Returns a new array which is a copy of the specified range of the original array.
 * @param fromIndex the start of the range (inclusive) to copy.
 * @param toIndex the end of the range (exclusive) to copy.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
public actual fun CharArray.copyOfRange(fromIndex: Int, toIndex: Int): CharArray {\n    this.asDynamic().slice(fromIndex, toIndex)\n\n/**
 * Fills this array or its subrange with the specified [element] value.
 * @param fromIndex the start of the range (inclusive) to fill, 0 by default.
 * @param toIndex the end of the range (exclusive) to fill, size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
@SinceKotlin("1.3")
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun <T> Array<T>.fill(element: T, fromIndex: Int = 0, toIndex: Int = size): Unit {\n    this.asDynamic().fill(element, fromIndex, toIndex)\n\n/**
 * Fills this array or its subrange with the specified [element] value.
 * @param fromIndex the start of the range (inclusive) to fill, 0 by default.
 * @param toIndex the end of the range (exclusive) to fill, size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
@SinceKotlin("1.3")
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun ByteArray.fill(element: Byte, fromIndex: Int = 0, toIndex: Int = size): Unit {\n    this.asDynamic().fill(element, fromIndex, toIndex)\n\n/**
 * Fills this array or its subrange with the specified [element] value.
 * @param fromIndex the start of the range (inclusive) to fill, 0 by default.
 * @param toIndex the end of the range (exclusive) to fill, size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
@SinceKotlin("1.3")
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun ByteArray.fill(element: Byte, fromIndex: Int = 0, toIndex: Int = size): Unit {\n    this.asDynamic().fill(element, fromIndex, toIndex)\n}
fromIndex the start of the range (inclusive) to fill, 0 by default.\n * @param toIndex the end of the range (exclusive) to fill, size of this array by default.\n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n
 */
@SinceKotlin("1.3")
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun ShortArray.fill(element: Short, fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n    this.asDynamic().fill(element, fromIndex, toIndex)\n}

/**
 * Fills this array or its subrange with the specified [element] value.\n * @param fromIndex the start of the range (inclusive) to fill, 0 by default.\n * @param toIndex the end of the range (exclusive) to fill, size of this array by default.\n * @param toIndex is less than zero or [toIndex] is greater than the size of this array.\n * @throws IndexOutOfBoundsException if [fromIndex] is greater than [toIndex].\n
 */
@SinceKotlin("1.3")
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun IntArray.fill(element: Int, fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n    this.asDynamic().fill(element, fromIndex, toIndex)\n}

/**
 * Fills this array or its subrange with the specified [element] value.\n * @param fromIndex the start of the range (inclusive) to fill, 0 by default.\n * @param toIndex the end of the range (exclusive) to fill, size of this array by default.\n * @param toIndex is less than zero or [toIndex] is greater than the size of this array.\n * @throws IndexOutOfBoundsException if [fromIndex] is greater than [toIndex].\n
 */
@SinceKotlin("1.3")
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun LongArray.fill(element: Long, fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n    this.asDynamic().fill(element, fromIndex, toIndex)\n}

/**
 * Fills this array or its subrange with the specified [element] value.\n * @param fromIndex the start of the range (inclusive) to fill, 0 by default.\n * @param toIndex the end of the range (exclusive) to fill, size of this array by default.\n * @param toIndex is less than zero or [toIndex] is greater than the size of this array.\n * @throws IndexOutOfBoundsException if [fromIndex] is greater than [toIndex].\n
 */
@SinceKotlin("1.3")
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun FloatArray.fill(element: Float, fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n    this.asDynamic().fill(element, fromIndex, toIndex)\n}

/**
 * Fills this array or its subrange with the specified [element] value.\n * @param fromIndex the start of the range (inclusive) to fill, 0 by default.\n * @param toIndex the end of the range (exclusive) to fill, size of this array by default.\n * @param toIndex is less than zero or [toIndex] is greater than the size of this array.\n * @throws IndexOutOfBoundsException if [fromIndex] is greater than [toIndex].\n
 */
@SinceKotlin("1.3")
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun DoubleArray.fill(element: Double, fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n    this.asDynamic().fill(element, fromIndex, toIndex)\n}

/**
 * Fills this array or its subrange with the specified [element] value.\n * @param fromIndex the start of the range (inclusive) to fill, 0 by default.\n * @param toIndex the end of the range (exclusive) to fill, size of this array by default.\n * @param toIndex is less than zero or [toIndex] is greater than the size of this array.\n * @throws IndexOutOfBoundsException if [fromIndex] is greater than [toIndex].\n
 */
@SinceKotlin("1.3")
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun BooleanArray.fill(element: Boolean, fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n    this.asDynamic().fill(element, fromIndex, toIndex)\n}
fromIndex the start of the range (inclusive) to fill, 0 by default.\n * @param toIndex the end of the range (exclusive) to fill, size of this array by default.\n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n
\n
```kotlin
@SinceKotlin("1.3")
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun CharArray.fill(element: Char, fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    this.asDynamic().fill(element, fromIndex, toIndex);
}

/**
 * Returns an array containing all elements of the original array and then the given [element].\n * @param toIndex the end of the range (exclusive) to fill, size of this array by default.\n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n
\n```
all elements of the original array and then all elements of the given [elements] collection.

*\n *\n *\n
public actual operator fun DoubleArray.plus(elements: Collection<Double>): DoubleArray {
    return fillFromCollection(this.copyOf(size + elements.size), this.size, elements)
}

public actual operator fun BooleanArray.plus(elements: Collection<Boolean>): BooleanArray {
    return arrayPlusCollection(this, elements)
}

public actual operator fun CharArray.plus(elements: Collection<Char>): CharArray {
    return fillFromCollection(this.copyOf(size + elements.size), this.size, elements)
}

public actual operator fun <T> Array<out T>.plus(elements: Array<out T>): Array<T> {
    return this.asDynamic().concat(elements)
}

public actual inline operator fun ByteArray.plus(elements: ByteArray): ByteArray {
    return primitiveArrayConcat(this, elements)
}

public actual inline operator fun ShortArray.plus(elements: ShortArray): ShortArray {
    return primitiveArrayConcat(this, elements)
}

public actual inline operator fun IntArray.plus(elements: IntArray): IntArray {
    return primitiveArrayConcat(this, elements)
}

public actual inline operator fun LongArray.plus(elements: LongArray): LongArray {
    return primitiveArrayConcat(this, elements)
}

public actual inline operator fun FloatArray.plus(elements: FloatArray): FloatArray {
    return primitiveArrayConcat(this, elements)
}

public actual inline operator fun DoubleArray.plus(elements: DoubleArray): DoubleArray {
    return primitiveArrayConcat(this, elements)
}

public actual inline operator fun BooleanArray.plus(elements: BooleanArray): BooleanArray {
    return primitiveArrayConcat(this, elements)
}

public actual inline operator fun CharArray.plus(elements: CharArray): CharArray {
    return primitiveArrayConcat(this, elements)
}

public actual inline fun <T> Array<out T>.plusElement(element: T): Array<T> {
    return this.asDynamic().concat(arrayOf(element))
}

public actual fun IntArray.sort(): Unit {
    definedExternally
}

public actual fun LongArray.sort(): Unit {
    if (size > 1) sort { a: Long, b: Long -> a.compareTo(b) }
}

public actual fun ByteArray.sort(): Unit {
    definedExternally
}

public actual fun ShortArray.sort(): Unit {
    definedExternally
}

public actual fun IntArray.sort(): Unit {
    definedExternally
}

public actual fun LongArray.sort(): Unit {
    if (size > 1) sort { a: Long, b: Long -> a.compareTo(b) }
}

public actual fun FloatArray.sort(): Unit {
    definedExternally
}

public actual fun DoubleArray.sort(): Unit {
    definedExternally
}

public actual fun BooleanArray.sort(): Unit {
    definedExternally
}

public actual fun CharArray.sort(): Unit {
    definedExternally
}
samples.collections.Arrays.Sorting.sortArrayOfComparable

/**
 * Sorts the array in-place according to the natural order of its elements.
 * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.
 * @sample samples.collections.Arrays.Sorting.sortArray
 */
@library("primitiveArraySort")
public actual fun <T : Comparable<T>> Array<out T>.sort(): Unit {
    if (size > 1) sortArray(this)
}

/**
 * Sorts a range in the array in-place.
 * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.
 * @sample samples.collections.Arrays.Sorting.sortRangeOfArrayOfComparable
 */
@SinceKotlin("1.4")
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun <T : Comparable<T>> Array<out T>.sort(fromIndex: Int = 0, toIndex: Int = size): Unit {
    val subarray = this.asDynamic().subarray(fromIndex, toIndex).unsafeCast<ByteArray>()
    subarray.sort()
}

/**
 * Sorts a range in the array in-place.
 * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.
 * @sample samples.collections.Arrays.Sorting.sortRangeOfByteArray
 */
@SinceKotlin("1.4")
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun ByteArray.sort(fromIndex: Int = 0, toIndex: Int = size): Unit {
    val subarray = this.asDynamic().subarray(fromIndex, toIndex).unsafeCast<ByteArray>()
    subarray.sort()
}

/**
 * Sorts a range in the array in-place.
 * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.
 * @sample samples.collections.Arrays.Sorting.sortRangeOfCharArray
 */
@SinceKotlin("1.4")
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun CharArray.sort(fromIndex: Int = 0, toIndex: Int = size): Unit {
    val subarray = this.asDynamic().subarray(fromIndex, toIndex).unsafeCast<CharArray>()
    subarray.sort()
}

/**
 * Sorts a range in the array in-place.
 * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.
 * @sample samples.collections.Arrays.Sorting.sortRangeOfCharSequence
 */
@SinceKotlin("1.4")
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun CharSequence.sort(fromIndex: Int = 0, toIndex: Int = size): Unit {
    val subarray = this.asDynamic().subarray(fromIndex, toIndex).unsafeCast<ByteArray>()
    subarray.sort()
}

/**
 * Sorts a range in the array in-place.
 * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.
 * @sample samples.collections.Arrays.Sorting.sortRangeOfString
 */
@SinceKotlin("1.4")
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun String.sort(fromIndex: Int = 0, toIndex: Int = size): Unit {
    val subarray = this.asDynamic().subarray(fromIndex, toIndex).unsafeCast<ByteArray>()
    subarray.sort()
}
range in the array in-place.

@throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.

@throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
fun DoubleArray.sort(noinline comparison: (a: Double, b: Double) -> Int): Unit {
    asDynamic().sort(comparison)
}

public inline fun DoubleArray.sort(noinline comparison: (a: Double, b: Double) -> Int): Unit {
    asDynamic().sort(comparison)
}

/**
 * Sorts the array in-place according to the order specified by the given [comparison] function.
 */
@kotlin.internal.InlineOnly
public inline fun CharArray.sort(noinline comparison: (a: Char, b: Char) -> Int): Unit {
    asDynamic().sort(comparison)
}

/**
 * Sorts the array in-place according to the order specified by the given [comparison] function.
 */
@kotlin.internal.InlineOnly
public inline fun CharArray.sort(noinline comparison: (a: Char, b: Char) -> Int): Unit {
    asDynamic().sort(comparison)
}

/**
 * Sorts the array in-place according to the order specified by the given [comparator].
 * The sort is _stable_. It means that equal elements preserve their order relative to each other
 * after sorting.
 * @param fromIndex the start of the range (inclusive) to sort, 0 by default.
 * @param toIndex the end of the range (exclusive) to sort, size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
@file:kotlin.jvm.JvmName("ComparisonsKt")
@file:kotlin.jvm.JvmMultifileClass

package kotlin.comparisons

/**
 * Compares two values using the specified functions [selectors] to calculate the result of the comparison.
 * As soon as the [Comparable] instances returned by a function for [a] and [b] values do not compare as equal, the result of that comparison is returned.
 * @sample samples.comparisons.Comparisons.compareValuesByWithSelectors
 */
public fun <T> compareValuesBy(a: T, b: T, vararg selectors: (T) -> Comparable<*>)

Comparable<*>?: Int {\n    return compareValues(selector(a), selector(b))\n}\n
/**
 * Compares two values
 * using the specified \[selector\] function to calculate the result of the comparison.\n * The function is applied to the given values \[a\] and \[b\] and return objects of type \[K\] which are then being\n * compared with the given \[comparator\].\n * Not so useful without type inference for receiver of expression\n */
public inline fun <T, K> compareValuesBy(a: T, b: T, comparator: Comparator<in K>, selector: (T) -> K): Int {\n    return compareValuesWith(a, b, comparator, selector, (T) -> K) {\n        if (a == null) return 0\n    }\
    if (b == null) return 1\n    if (a < b) return -1\n    if (a > b) return 1\n}\n
/**
 * Creates a comparator using the sequence of functions to calculate a
 * result of comparison.\n * The functions are called sequentially, receive the given values `a` and `b` and return
 * \[Comparable\] objects. As soon as the \[Comparable\] instances returned by a function for `a` and `b` values do
 * not compare as equal, the result of that comparison is returned from the \[Comparator\].\n */
public fun <T : Comparable<*>> compareValues(a: T?, b: T?): Int {\n    if (a === b) return 0\n    if (a == null) return -1\n    if (b == null) return 1\n    @Suppress("UNCHECKED_CAST")\n    return (a as Comparable<Any>).compareTo(b)\n}\n
/**
 * Creates a comparator using the given \[selector\] function to transform values
 * being compared and then applying\n * the specified \[comparator\] to compare transformed values.\n */
public fun <T, K> compareBy(comparator: Comparator<in K>, crossinline selector: (T) -> K): Comparator<T> =
    Comparator { a, b -> compareValuesBy(a, b, comparator, selector) }

/**
 * Creates a comparator comparing values after the primary comparator defined them equal. It
 * uses\n * the function to transform values to a \[Comparable\] instance for comparison.\n * Not that an order of \[comparator\] is
 * reversed by this wrapper.\n */
public inline fun <T> Comparator<T>.thenBy(crossinline selector: (T) -> Comparable<*>?): Comparator<T> =
    Comparator { a, b ->
        val previousCompare = this@thenBy.compare(a, b)\n        if (previousCompare != 0) previousCompare else
            compareValuesBy(a, b, selector)\n    }\n
Open Source Used In security_automation identity-guest-cs 1391
this@thenBy.compare(a, b)/*n */(if (previousCompare != 0) previousCompare else compareValuesBy(a, b, comparator, selector))/*n */*n * Creates a descending comparator using the primary comparator and the function to transform value to a [Comparable] instance for comparison./*n */*n * @sample samples.comparisons.Comparisons.thenByDescending/*n */*/@kotlin.internal.InlineOnly\n/public inline fun <!T> Comparator<!T>: thenByDescending( crossinline selector: (T) -> Comparable<*?>): Comparator<!T> =*/n Comparator { a, b ->/*n val previousCompare = this@thenByDescending.compare(a, b)/*n if (previousCompare != 0) previousCompare else compareValuesBy(b, a, selector))/*n */*n */*n * Creates a descending comparator comparing values after the primary comparator defined them equal. It uses the [selector] function to transform values and then compares them with the given [comparator]./*n */*n * @sample samples.comparisons.Comparisons.thenByDescendingWithComparator/*n */*/@kotlin.internal.InlineOnly\n/public inline fun <T, K> Comparator<!T>: thenByDescending( comparator: Comparator<in K>, crossinline selector: (T) -> K): Comparator<!T> =*/n Comparator<!T> =*/n Comparator { a, b ->/*n val previousCompare = this@thenByDescending.compare(a, b)/*n if (previousCompare != 0) previousCompare else compareValuesBy(b, a, selector))/*n */*n */*n * Creates a comparator using the primary comparator and function to calculate a result of comparison./*n */*n * @sample samples.comparisons.Comparisons.thenComparator/*n */*/@kotlin.internal.InlineOnly\n/public inline fun <T> Comparator<!T>: thenComparator( crossinline comparison: (a: T, b: T) -> Int): Comparator<!T> =*/n Comparator { a, b ->/*n val previousCompare = this@thenComparator.compare(a, b)/*n if (previousCompare != 0) previousCompare else comparison(a, b)/*n */*n */*n * Extends the given [comparator] of non-nullable values to a comparator of nullable values considering ‘null’ value less than any other value./*n */*n * @sample samples.comparisons.Comparisons.thenFirstLastWithComparator/*n */*/@kotlin.internal.InlineOnly\n/public inline fun <!T>: Any> nullsFirst( comparator: Comparator<!T>): Comparator<!T> =*/n Comparator { a, b ->/*n when {/*n a == b -> 0/*n a == null -> -1/*n b == null -> 1/*n else -> comparator.compare(a, b)/*n }/*n }/*n */*n * Extends the given [comparator] of non-nullable values to a comparator of nullable values considering ‘null’ value less than any other value./*n */*n * @sample samples.comparisons.Comparisons.thenFirstLastWithComparator/*n */*/@kotlin.internal.InlineOnly\n/public inline fun <!T>: Any> nullsLast( comparator: Comparator<!T>): Comparator<!T> =*/n Comparator { a, b ->/*n when {/*n a == b -> 0/*n a == null -> 1/*n b == null -> -1/*n else -> comparator.compare(a, b)/*n }/*n }/*n */*n * Extends the given [comparator] of non-nullable values to a comparator of nullable values considering ‘null’ value greater than any other value./*n */*n * @sample samples.comparisons.Comparisons.thenFirstLastWithComparator/*n */*/@kotlin.internal.InlineOnly\n/public inline fun <!T>: Any> nullsFirst(): Comparator<!T> =*/n nullsFirst( naturalOrder())/*n */*n */*n * Returns a comparator that compares [Comparable] objects in natural order./*n */*n * @sample samples.comparisons.Comparisons.naturalOrderComparator/*n */*/@kotlin.internal.InlineOnly\n/public inline fun <!T>: Any> nullsLast(): Comparator<!T> =*/n nullsLast( naturalOrder())/*n */*n */*n * Returns a comparator that compares [Comparable] objects in reversed natural order./*n */*n * @sample samples.comparisons.Comparisons.naturalOrderComparator
Comparable&lt;T&gt;::reverseOrder(): Comparator&lt;T&gt; = @Suppress("UNCHECKED_CAST")
(ReverseOrderComparator as Comparator&lt;T&gt;)&

/**
 *  Returns a comparator that imposes the reverse ordering
 *  of this comparator.
 *
 *  @sample samples.comparisons.Comparisons.reversed
 */
public fun &lt;T&gt; Comparator&lt;T&gt;.reversed(): Comparator&lt;T&gt; = when (this) {
    is ReversedComparator -> this.comparator
    NaturalOrderComparator -> @Suppress("UNCHECKED_CAST") (ReverseOrderComparator as

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language
 * contributors.
 */

@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("StandardKt")
package kotlin

import kotlin.contracts.*

/**
 * An exception is thrown to indicate that a method body remains to be implemented.
 */
public class NotImplementedError(message: String = "An operation is not implemented.") :

/**
 * Always throws [NotImplementedError] stating that operation is not implemented.
 */
@kotlin.internal.InlineOnly
public inline fun TODO(): Nothing = throw NotImplementedError()

/**
 * Always throws [NotImplementedError] stating that operation is not implemented. and
 * param reason a string explaining why the implementation is missing.
 */
@kotlin.internal.InlineOnly
public inline fun TODO(reason: String): Nothing = throw NotImplementedError("An operation is not implemented: $reason")

/**
 * Calls the specified function [block] and returns its result.
 */
@kotlin.internal.InlineOnly
public inline fun &lt;R&gt; run(block: () -> R): R {
    contract {
        callsInPlace(block, InvocationKind.EXACTLY_ONCE)
    }
    return block()
}

/**
 * Calls the specified function [block] with `this` value as its receiver and returns its result.
 */
@kotlin.internal.InlineOnly
@SinceKotlin("1.1")
public inline fun &lt;T&gt; T.apply(block: T.() -> Unit): T {
    contract {
        callsInPlace(block, InvocationKind.EXACTLY_ONCE)
    }
    block()
    return this
}

/**
 */

For detailed usage information see the documentation for [scope functions](https://kotlinlang.org/docs/reference/scope-functions.html#let).

```kotlin
@kotlin.internal.InlineOnly
public inline fun <T, R> T.let(block: (T) -> R): R {
    contract {
        callsInPlace(block, InvocationKind.EXACTLY_ONCE)
    } return block(this)
}
```

```kotlin
@kotlin.internal.InlineOnly
public inline fun <T> T.takeIf(predicate: (T) -> Boolean): T? {
    contract {
        callsInPlace(predicate, InvocationKind.EXACTLY_ONCE)
    } return if (predicate(this)) this else null
}
```

```kotlin
@kotlin.internal.InlineOnly
public inline fun <T> T.takeUnless(predicate: (T) -> Boolean): T? {
    contract {
        callsInPlace(predicate, InvocationKind.EXACTLY_ONCE)
    } return if (!predicate(this)) this else null
}
```

```kotlin
@kotlin.internal.InlineOnly
public inline fun repeat(times: Int, action: (Int) -> Unit) {
    contract { callsInPlace(action) }
    for (index in 0 until times) {
        action(index)
    }
}
```

```kotlin
@SinceKotlin("1.1")
public actual fun <T : Comparable<T>> maxOf(a: T, b: T): T {
    return if (a >= b) a else b
}
```

```kotlin
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
@Suppress("DEPRECATION_ERROR")
public actual inline fun maxOf(a: Byte, b: Byte): Byte {
    return Math.max(a.toInt(), b.toInt()).unsafeCast<Byte>()
}
```

```kotlin
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
@Suppress("DEPRECATION_ERROR")
public actual inline fun maxOf(a: Short, b: Short): Short {
    return Math.max(a.toInt(), b.toInt()).unsafeCast<Short>()
}
```

```kotlin
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
@Suppress("DEPRECATION_ERROR")
public actual inline fun maxOf(a: Int, b: Int): Int {
    return Math.max(a, b)
}
```

```kotlin
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
@Suppress("DEPRECATION_ERROR")
public actual inline fun maxOf(a: Long, b: Long): Long {
    return if (a >= b) a else b
}
```

```kotlin
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
@Suppress("DEPRECATION_ERROR")
public actual inline fun maxOf(a: Float, b: Float): Float {
    return Math.max(a, b).toInt()
}
```

```kotlin
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
@Suppress("DEPRECATION_ERROR")
public actual inline fun maxOf(a: Short, b: Short): Short {
    return Math.max(a.toInt(), b.toInt()).toInt()
}
```

```kotlin
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
@Suppress("DEPRECATION_ERROR")
public actual inline fun maxOf(a: Int, b: Int): Int {
    return Math.max(a, b)
}
```

```kotlin
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
@Suppress("DEPRECATION_ERROR")
public actual inline fun maxOf(a: Long, b: Long): Long {
    return Math.max(a, b)
}
```

```kotlin
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
@Suppress("DEPRECATION_ERROR")
public actual inline fun maxOf(a: Float, b: Float): Float {
    return Math.max(a, b)
}
```

```kotlin
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
@Suppress("DEPRECATION_ERROR")
public actual inline fun maxOf(a: Short, b: Short): Short {
    return Math.max(a, b)
}
```
c.toInt()).unsafeCast<Short>() * Returns the greater of three values.

@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
@Suppress("DEPRECATION_ERROR")
public actual inline fun maxOf(a: Int, b: Int, c: Int): Int {
    return Math.max(a, b, c)
} * Returns the greater of three values.

@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
@Suppress("DEPRECATION_ERROR")
public actual inline fun maxOf(a: Int, b: Int, c: Int): Int {
    return Math.max(a, b, c)
} * Returns the greater of three values.

@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
@Suppress("DEPRECATION_ERROR")
public actual inline fun maxOf(a: Float, b: Float, c: Float): Float {
    return Math.max(a, b, c)
} * Returns the greater of three values.

@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
@Suppress("DEPRECATION_ERROR")
public actual inline fun maxOf(a: Long, b: Long, c: Long): Long {
    return Math.max(a, maxOf(b, c))
} * Returns the greater of three values.

If any value is `NaN`, returns `NaN`.

@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
@Suppress("DEPRECATION_ERROR")
public actual inline fun maxOf(a: Byte, b: Byte): Byte {
    return Math.min(a.toInt(), b.toInt()).unsafeCast<Byte>()
} * Returns the greater of two values.

@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
@Suppress("DEPRECATION_ERROR")
public actual inline fun maxOf(a: Short, b: Short): Short {
    return Math.min(a.toInt(), b.toInt()).unsafeCast<Short>()
} * Returns the greater of two values.

@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
@Suppress("DEPRECATION_ERROR")
public actual inline fun maxOf(a: Int, b: Int): Int {
    return Math.min(a, b)
} * Returns the smaller of two values.

@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
@Suppress("DEPRECATION_ERROR")
public actual inline fun maxOf(a: Long, b: Long): Long {
    return if (a <= b) a else b
} * Returns the smaller of two values.

@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
@Suppress("DEPRECATION_ERROR")
public actual inline fun minOf(a: Float, b: Float): Float {
    return Math.min(a, b)
} * Returns the smaller of two values.

@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
@Suppress("DEPRECATION_ERROR")
public actual inline fun minOf(a: Double, b: Double): Double {
    return Math.min(a, b)
} * Returns the smaller of two values.

@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
@Suppress("DEPRECATION_ERROR")
public actual inline fun minOf(a: Float, b: Float): Float {
    return Math.min(a, b)
} * Returns the smaller of two values.

@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
@Suppress("DEPRECATION_ERROR")
public actual inline fun minOf(a: Double, b: Double): Double {
    return Math.min(a, b)
} * Returns the smaller of two values.

@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
@Suppress("DEPRECATION_ERROR")
public actual inline fun minOf(a: Float, b: Float): Float {
    return Math.min(a, b)
} * Returns the smaller of two values.

@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
@Suppress("DEPRECATION_ERROR")
public actual inline fun minOf(a: Double, b: Double): Double {
    return Math.min(a, b)
} * Returns the smaller of two values.

@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
@Suppress("DEPRECATION_ERROR")
public actual inline fun minOf(a: Float, b: Float): Float {
    return Math.min(a, b)
} * Returns the smaller of two values.

@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
@Suppress("DEPRECATION_ERROR")
public actual inline fun minOf(a: Double, b: Double): Double {
    return Math.min(a, b)
} * Returns the smaller of two values.

@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
@Suppress("DEPRECATION_ERROR")
public actual inline fun minOf(a: Float, b: Float): Float {
    return Math.min(a, b)
} * Returns the smaller of two values.
@kotlin.internal.InlineOnly
public inline operator fun compareTo(other: UShort): Int = 
this.compareTo(other.toULong())
/**
 * Compares this value with the specified value for order.
 * Returns zero if this value is equal to the specified other value, a negative number if it's less than other,
 * or a positive number if it's greater than other.
 */
@kotlin.internal.InlineOnly
public inline operator fun compareTo(other: UInt): Int = 
this.compareTo(other.toULong())
/**
 * Compares this value with the specified value for order.
 * Returns zero if this value is equal to the specified other value, a negative number if it's less than other,
 * or a positive number if it's greater than other.
 */
@kotlin.internal.InlineOnly
public inline operator fun compareTo(other: ULong): Int = 
ulongCompare(this.data, other.data)
/** Adds the other value to this value.
 */
@kotlin.internal.InlineOnly
public inline operator fun plus(other: UByte): ULong = this.plus(other.toULong())
/** Adds the other value to this value.
 */
@kotlin.internal.InlineOnly
public inline operator fun plus(other: UShort): ULong = this.plus(other.toULong())
/** Adds the other value to this value.
 */
@kotlin.internal.InlineOnly
public inline operator fun plus(other: UInt): ULong = this.plus(other.toULong())
/** Adds the other value to this value.
 */
@kotlin.internal.InlineOnly
public inline operator fun plus(other: ULong): ULong = ULong(this.data.plus(other.data))
/** Subtracts the other value from this value.
 */
@kotlin.internal.InlineOnly
public inline operator fun minus(other: UByte): ULong = this.minus(other.toULong())
/** Subtracts the other value from this value.
 */
@kotlin.internal.InlineOnly
public inline operator fun minus(other: UShort): ULong = this.minus(other.toULong())
/** Subtracts the other value from this value.
 */
@kotlin.internal.InlineOnly
public inline operator fun minus(other: UInt): ULong = this.minus(other.toULong())
/** Subtracts the other value from this value.
 */
@kotlin.internal.InlineOnly
public inline operator fun minus(other: ULong): ULong = ulongDivide(this, other)
/** Multiplies this value by the other value.
 */
@kotlin.internal.InlineOnly
public inline operator fun times(other: UByte): ULong = this.times(other.toULong())
/** Multiplies this value by the other value.
 */
@kotlin.internal.InlineOnly
public inline operator fun times(other: UShort): ULong = this.times(other.toULong())
/** Multiplies this value by the other value.
 */
@kotlin.internal.InlineOnly
public inline operator fun times(other: UInt): ULong = this.times(other.toULong())
/** Multiplies this value by the other value.
 */
@kotlin.internal.InlineOnly
public inline operator fun times(other: ULong): ULong = ULong(this.data.times(other.data))
/** Divides this value by the other value.
 */
@kotlin.internal.InlineOnly
public inline operator fun div(other: UByte): ULong = this.div(other.toULong())
/** Divides this value by the other value.
 */
@kotlin.internal.InlineOnly
public inline operator fun div(other: UShort): ULong = this.div(other.toULong())
/** Divides this value by the other value.
 */
@kotlin.internal.InlineOnly
public inline operator fun div(other: UInt): ULong = this.div(other.toULong())
/** Divides this value by the other value.
 */
@kotlin.internal.InlineOnly
public inline operator fun div(other: ULong): ULong = ulongDivide(this, other)
/** Calculates the remainder of dividing this value by the other value.
 */
@kotlin.internal.InlineOnly
public inline operator fun rem(other: UByte): ULong = this.rem(other.toULong())
/** Calculates the remainder of dividing this value by the other value.
 */
@kotlin.internal.InlineOnly
public inline operator fun rem(other: UShort): ULong = this.rem(other.toULong())
/** Calculates the remainder of dividing this value by the other value.
 */
@kotlin.internal.InlineOnly
public inline operator fun rem(other: UInt): ULong = this.rem(other.toULong())
/** Calculates the remainder of dividing this value by the other value.
 */
@kotlin.internal.InlineOnly
public inline operator fun rem(other: ULong): ULong = ulongRemainder(this, other)
/** Creates a range from this value to the specified [other] value.
 */
@kotlin.internal.InlineOnly
public inline operator fun rangeTo(other: ULong): ULongRange = ULongRange(this, other)
/** Shifts this value left by the [bitCount] number of bits.
 */
@kotlin.internal.InlineOnly
public inline operator fun shl(bitCount: Int): ULong = ULong(data shl bitCount)
/**
 * Shifts this value right by the [bitCount] number of bits, filling the leftmost bits with zeros.
 * Note that only the six lowest-order bits of the [bitCount] are used as the shift distance.
 * The shift distance actually used is therefore always in the range `0..63`.
 */
@kotlin.internal.InlineOnly
public inline infix fun shr(bitCount: Int): ULong = ULong(data.ushr(bitCount))

/**
 * Performs a bitwise AND operation between the two values.
 */
@kotlin.internal.InlineOnly
public inline infix fun and(other: ULong): ULong = ULong(this.data and other.data)

/**
 * Performs a bitwise OR operation between the two values.
 */
@kotlin.internal.InlineOnly
public inline infix fun or(other: ULong): ULong = ULong(this.data xor other.data)

/**
 * Performs a bitwise XOR operation between the two values.
 */
@kotlin.internal.InlineOnly
public inline infix fun xor(other: ULong): ULong = ULong(this.data xor other.data)

/**
 * Inverts the bits in this value.
 */
@kotlin.internal.InlineOnly
public inline infix fun inv(): ULong = ULong(data.inv())

/**
 * Converts this [ULong] value to [Byte].
 */
@kotlin.internal.InlineOnly
public inline fun toByte(): Byte = data.toByte()

/**
 * Converts this [ULong] value to [UByte].
 */
@kotlin.internal.InlineOnly
public inline fun toUByte(): UByte = data.toUByte()

/**
 * Converts this [ULong] value to [UShort].
 */
@kotlin.internal.InlineOnly
public inline fun toUShort(): UShort = data.toUShort()

/**
 * Converts this [ULong] value to [UInt].
 */
@kotlin.internal.InlineOnly
public inline fun toUInt(): UInt = data.toUInt()

/**
 * Converts this [ULong] value to [Int].
 */
@kotlin.internal.InlineOnly
public inline fun toInt(): Int = data.toInt()

/**
 * Converts this [ULong] value to [Long].
 */
@kotlin.internal.InlineOnly
public inline fun toLong(): Long = data

/**
 * Converts this [ULong] value to [Float].
 */
@kotlin.internal.InlineOnly
public inline fun toFloat(): Float = this.toDouble().toFloat()

/**
 * Converts this [ULong] value to [Double].
 */
@kotlin.internal.InlineOnly
public inline fun toDouble(): Double = ulongToDouble(data)

/**
 * Converts this [Byte] value to [ULong].
 */
@kotlin.internal.InlineOnly
public inline fun fromByte(): ULong = ulongFromByte(data)

/**
 * Converts this [ULong] value to [Byte].
 */
@kotlin.internal.InlineOnly
public inline fun fromULong(): Byte = data.toByte()

/**
 * Converts this [ULong] value to [UByte].
 */
@kotlin.internal.InlineOnly
public inline fun fromUByte(): UByte = data.toUByte()

/**
 * Converts this [ULong] value to [UShort].
 */
@kotlin.internal.InlineOnly
public inline fun fromUShort(): UShort = data.toUShort()

/**
 * Converts this [ULong] value to [UInt].
 */
@kotlin.internal.InlineOnly
public inline fun fromUInt(): UInt = data.toUInt()

/**
 * Converts this [ULong] value to [Int].
 */
@kotlin.internal.InlineOnly
public inline fun fromInt(): Int = data.toInt()

/**
 * Converts this [ULong] value to [Long].
 */
@kotlin.internal.InlineOnly
public inline fun fromLong(): Long = data

/**
 * Converts this [ULong] value to [Float].
 */
@kotlin.internal.InlineOnly
public inline fun fromFloat(): Float = this.toDouble().toFloat()

/**
 * Converts this [ULong] value to [Double].
 */
@kotlin.internal.InlineOnly
public inline fun fromDouble(): Double = ulongToDouble(data)

/**
 * Converts this [Byte] value to [ULong].
 */
@kotlin.internal.InlineOnly
public inline fun fromByte(): ULong = ulongFromByte(data)

/**
 * Converts this [ULong] value to [Byte].
 */
@kotlin.internal.InlineOnly
public inline fun fromULong(): Byte = data.toByte()

/**
 * Converts this [ULong] value to [UByte].
 */
@kotlin.internal.InlineOnly
public inline fun fromUByte(): UByte = data.toUByte()

/**
 * Converts this [ULong] value to [UShort].
 */
@kotlin.internal.InlineOnly
public inline fun fromUShort(): UShort = data.toUShort()

/**
 * Converts this [ULong] value to [UInt].
 */
@kotlin.internal.InlineOnly
public inline fun fromUInt(): UInt = data.toUInt()

/**
 * Converts this [ULong] value to [Int].
 */
@kotlin.internal.InlineOnly
public inline fun fromInt(): Int = data.toInt()

/**
 * Converts this [ULong] value to [Long].
 */
@kotlin.internal.InlineOnly
public inline fun fromLong(): Long = data

/**
 * Converts this [ULong] value to [Float].
 */
@kotlin.internal.InlineOnly
public inline fun fromFloat(): Float = this.toDouble().toFloat()

/**
 * Converts this [ULong] value to [Double].
 */
@kotlin.internal.InlineOnly
public inline fun fromDouble(): Double = ulongToDouble(data)

/**
 * Converts this [Byte] value to [ULong].
 */
@kotlin.internal.InlineOnly
public inline fun fromByte(): ULong = ulongFromByte(data)

/**
 * Converts this [ULong] value to [Byte].
 */
@kotlin.internal.InlineOnly
public inline fun fromULong(): Byte = data.toByte()

/**
 * Converts this [ULong] value to [UByte].
 */
@kotlin.internal.InlineOnly
public inline fun fromUByte(): UByte = data.toUByte()

/**
 * Converts this [ULong] value to [UShort].
 */
@kotlin.internal.InlineOnly
public inline fun fromUShort(): UShort = data.toUShort()

/**
 * Converts this [ULong] value to [UInt].
 */
@kotlin.internal.InlineOnly
public inline fun fromUInt(): UInt = data.toUInt()

/**
 * Converts this [ULong] value to [Int].
 */
@kotlin.internal.InlineOnly
public inline fun fromInt(): Int = data.toInt()

/**
 * Converts this [ULong] value to [Long].
 */
@kotlin.internal.InlineOnly
public inline fun fromLong(): Long = data

/**
 * Converts this [ULong] value to [Float].
 */
@kotlin.internal.InlineOnly
public inline fun fromFloat(): Float = this.toDouble().toFloat()

/**
 * Converts this [ULong] value to [Double].
 */
@kotlin.internal.InlineOnly
public inline fun fromDouble(): Double = ulongToDouble(data)

/**
 * Converts this [Byte] value to [ULong].
 */
@kotlin.internal.InlineOnly
public inline fun fromByte(): ULong = ulongFromByte(data)

/**
 * Converts this [ULong] value to [Byte].
 */
@kotlin.internal.InlineOnly
public inline fun fromULong(): Byte = data.toByte()

/**
 * Converts this [ULong] value to [UByte].
 */
@kotlin.internal.InlineOnly
public inline fun fromUByte(): UByte = data.toUByte()

/**
 * Converts this [ULong] value to [UShort].
 */
@kotlin.internal.InlineOnly
public inline fun fromUShort(): UShort = data.toUShort()

/**
 * Converts this [ULong] value to [UInt].
 */
@kotlin.internal.InlineOnly
public inline fun fromUInt(): UInt = data.toUInt()

/**
 * Converts this [ULong] value to [Int].
 */
@kotlin.internal.InlineOnly
public inline fun fromInt(): Int = data.toInt()
significant 56 bits are filled with the sign bit of this value.

```kotlin
*\@Since Kotlin("1.3.0") *\@ExperimentalUnsignedTypes *\@kotlin.internal.InlineOnly *\@unpublic inline fun Byte.toULong(): ULong = ULong(this.toLong())
```

Converts this [Short] value to [ULong].

*\@Since Kotlin("1.3.0") *\@ExperimentalUnsignedTypes *\@kotlin.internal.InlineOnly *\@unpublic inline fun Short.toULong(): ULong = ULong(this.toLong())

Converts this [Int] value to [ULong].

*\@Since Kotlin("1.3.0") *\@ExperimentalUnsignedTypes *\@kotlin.internal.InlineOnly *\@unpublic inline fun Int.toULong(): ULong = ULong(this.toLong())

Converts this [Long] value to [ULong].

*\@Since Kotlin("1.3.0") *\@ExperimentalUnsignedTypes *\@kotlin.internal.InlineOnly *\@unpublic inline fun Long.toULong(): ULong = ULong(this.toLong())

Converts this [Float] value to [ULong].

*\@Since Kotlin("1.3.0") *\@ExperimentalUnsignedTypes *\@kotlin.internal.InlineOnly *\@unpublic inline fun Float.toULong(): ULong = doubleToULong(this.toDouble())

Converts this [Double] value to [ULong].

```

@file: kotlin.jvm.JvmMultifileClass
@file: kotlin.jvm.JvmName("CollectionsKt")

package kotlin.collections

// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt
// See: https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib

import kotlin.random.*
import kotlin.ranges.contains
import kotlin.ranges.reversed

/**
 * Returns 1st *element* from the list.
 * 
 * Throws an [IndexOutOfBoundsException] if the size of this list is less than 1.
 */
@kotlin.internal.InlineOnly
public inline operator fun <T> List<T>.component1(): T {
    return get(0)
}

/**
 * Returns 2nd *element* from the list.
 * 
 * Throws an [IndexOutOfBoundsException] if the size of this list is less than 2.
 */
@kotlin.internal.InlineOnly
public inline operator fun <T> List<T>.component2(): T {
    return get(1)
}

/**
 * Returns 3rd *element* from the list.
 * 
 * Throws an [IndexOutOfBoundsException] if the size of this list is less than 3.
 */
@kotlin.internal.InlineOnly
public inline operator fun <T> List<T>.component3(): T {
    return get(2)
}

/**
 * Returns 4th *element* from the list.
 * 
 * Throws an [IndexOutOfBoundsException] if the size of this list is less than 4.
 */
@kotlin.internal.InlineOnly
public inline operator fun <T> List<T>.component4(): T {
    return get(3)
}

/**
 * Returns 5th *element* from the list.
 * 
 * Throws an [IndexOutOfBoundsException] if the size of this list is less than 5.
 */
@kotlin.internal.InlineOnly
public inline operator fun <T> List<T>.component5(): T {
    return get(4)
}

/**
 * Returns an [IndexOutOfBoundsException] if the size of this list is less than 1.
 */
public operator fun <@kotlin.internal.OnlyInputTypes T> Iterable<T>.contains(element: T): Boolean {
    if (this is Collection)
        return contains(element)
    return indexOf(element) >= 0
}

/**
 * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the size of this list is less than 4.
 */
public fun <T> Iterable<T>.elementAt(index: Int): T {
    if (this is List)
        return get(index)
    return
```
elementAtOrElse(index) { throw IndexOutOfBoundsException("Collection doesn't contain element at index ": index)) } Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this list.

get(index) { return defaultValue(index) } Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this list.

forEach(sample, fun) { sample } Iterables.forEach(sample) Returns a result of calling the [fun] function on each element of the collection.

first() Returns the first element of the collection.

firstOrNull() Returns the first element, or null if the collection is empty.

find(predicate) Returns the first element matching the given [predicate], or null if no such element was found.

findLast(predicate) Returns the last element matching the given [predicate], or null if no such element was found.

forEach() Performs an action for each element of the collection.

indexOf(element) Returns the index of the first occurrence of the given element, or -1 if the element is not found.

lastOrDefault(element, defaultValue) Returns the last element, or the result of calling the [defaultValue] function if the element is not found.

lastIndex() Returns the index of the last element.

lastOrNull() Returns the last element, or null if the list is empty.

find(predicate) Returns the first element matching the given [predicate], or null if no such element was found.

get(index) Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this list.

iterator() Returns an iterable representing the elements of the collection.

isEmpty() Returns true if the list is empty, false otherwise.

remove Keyed Collections.remove(key) Removes the element associated with the given key.

set(index, element) Sets the element at the given [index] to the new value.

size() Returns the number of elements in the collection.

toArrayList() Converts the collection to an ArrayList.

toHashSet() Converts the collection to a HashSet.

toMap(keySelector, valueSelector) Converts the collection to a Map.

toArray() Converts the collection to an Array.

values() Returns an iterable containing the elements of the collection.
Open Source Used In security_automation identity-guest-cs 1401

```kotlin
@kotlin.internal.InlineOnly
public fun <@kotlin.internal.OnlyInputTypes T> Iterable<T>.indexOf(element: T): Int {
    if (this is List)
        return this.indexOf(element)
    var index = 0
    for (item in this) {
        checkIndexOverflow(index)
        if (element == item)
            lastIndex = index
        index++
    }
    return lastIndex
}

/**
 * Returns index of the first element matching the given [predicate], or -1 if the collection does not contain such element.
 */
@Suppress("UNCHECKED_CAST")
public fun <@kotlin.internal.OnlyInputTypes T> Iterable<T>.indexOfFirst(predicate: (T) -> Boolean): Int {
    var index = 0
    for (item in this) {
        if (predicate(item))
            return index
        index++
    }
    return -1
}

/**
 * Returns index of the first element matching the given [predicate].
 */
@Suppress("EXTENSION_SHADOWED_BY_MEMBER") // false warning, extension takes precedence in some cases
@kotlin.internal.InlineOnly
public inline fun <T> List<T>.indexOfFirst(predicate: (T) -> Boolean): Int {
    var index = 0
    for (item in this) {
        if (predicate(item))
            return index
        index++
    }
    return -1
}

/**
 * Returns index of the last element matching the given [predicate], or -1 if the collection does not contain such element.
 */
public inline fun <T> List<T>.indexOfLast(predicate: (T) -> Boolean): Int {
    var lastIndex = -1
    var index = 0
    for (item in this) {
        checkIndexOverflow(index)
        if (element
```
Returns last index of [element], or -1 if the list does not contain element.

Returns the last element, or `null` if the collection is empty. Returns the last element matching the given [predicate], or `null` if no such element was found.

Returns the last element matching the given [predicate], or `null` if no such element was found.

Returns a random element from this collection.

Returns a random element from this collection using the specified source of randomness.

Returns a random element from this collection.

Returns the single element, or throws an exception if there is no or more than one matching element.

Returns the single element, or throws an exception if the list is empty or has more than one element.

Returns the single element, or throws an exception if the collection is empty or has more than one element.

Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element.

Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element.

Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element.

Returns the single element.
*/
 * @sample samples.collections.Collections.Transformations.drop
 */
public fun <T> Iterable<T>.drop(n: Int): List<T> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return toList()
    val list: ArrayList<T>
    if (this is Collection<*>) {
        val resultSize = size - n
        if (resultSize <= 0)
            return emptyList()
        if (resultSize == 1)
            return listOf(last())
        list = ArrayList<T>(resultSize)
        if (this is List<T>) {
            if (this is RandomAccess) {
                for (index in n until size)
                    list.add(this[index])
            } else {
                for (item in listIterator(n))
                    list.add(item)
            }
        return list
    }
    else {
        list = ArrayList<T>()
    }
    var count = 0
    for (item in this) {
        if (count >= n) list.add(item) else ++count
    }
    return list.optimizeReadOnlyList()
}

/**
 * Returns a list containing all elements except first 
[n] elements.
 * 
 * @throws IllegalArgumentException if 
[n] is negative.
 * @sample samples.collections.Collections.Transformations.drop
 */
public fun <T> List<T>.dropLast(n: Int): List<T> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return take((size - n).coerceAtLeast(0))
}

/**
 * Returns a list containing all elements except last 
[n] elements that satisfy the given 
[predicate].
 * 
 * @sample samples.collections.Collections.Transformations.drop
 */
public inline fun <T> List<T>.dropLastWhile(predicate: (T) -> Boolean): List<T> {
    if (!isEmpty()) {
        val iterator = listIterator(size)
        while (iterator.hasPrevious()) {
            if (!predicate(iterator.previous())) {
                return take(iterator.nextIndex() + 1)
            }
        }
    }
    return emptyList()
}

/**
 * Returns a list containing all elements except first elements that satisfy the given 
[predicate].
 * 
 * @sample samples.collections.Collections.Transformations.drop
 */
public inline fun <T> Iterable<T>.dropWhile(predicate: (T) -> Boolean): List<T> {
    var yielding = false
    val list = ArrayList<T>()
    for (item in this)
        if (yielding)
            list.add(item)
        else if (!predicate(item)) {
            list.add(item)
            yielding = true
        }
    return list.optimizeReadOnlyList()}

/**
 * Returns a list containing elements matching the given 
[predicate].
 * 
 * @sample samples.collections.Collections.Transformations.drop
 */
public inline fun <T> Iterable<T>.filter(predicate: (T) -> Boolean): List<T> {
    return filterTo(ArrayList<T>(), predicate)
}

/**
 * Returns a list containing elements matching the given 
[predicate].
 * 
 * @sample samples.collections.Collections.Transformations.drop
 */
public inline fun <T> Iterable<T>.filterIndexed(predicate: (index: Int, T) -> Boolean): List<T> {
    return filterIndexedTo(ArrayList<T>(), predicate)
}

/**
 * Appends all elements matching the given 
[predicate] to
 * the given 
[destination].
 * 
 * @sample samples.collections.Collections.Transformations.drop
 */
public inline fun <T, C : MutableCollection<in T>> Iterable<T>.filterIndexedTo(destination: C, predicate: (index: Int, T) -> Boolean): C {
    forEachIndexed { index, element ->
        if (predicate(index, element)) destination.add(element)
    }
    return destination
}

/**
 * Returns a list containing all elements that are instances of specified type parameter 
[R].
 * 
 * @sample samples.collections.Collections.Transformations.drop
 */
public inline fun <reified R> Iterable<*>.filterIsInstance(): List<R> {
    return filterIsInstanceTo(ArrayList<R>())
}

/**
 * Returns a list containing all elements that are instances of specified type parameter 
[R].
 * 
 * @sample samples.collections.Collections.Transformations.drop
 */
public inline fun <reified R, C : MutableCollection<in R>> Iterable<*>.filterIsInstanceTo(destination: C): C {
    return filterIsInstanceTo(destination)
}

Open Source Used In security_automation identity-guest-cs 1403
R) destination.add(element)\n    return destination\n\n* Returns a list containing all elements not matching
the given [predicate].\n* in * @sample samples.collections.Collections.Filtering.filter.*in\n* in * FilterNot(predicate: (T) -> Boolean): List\n\n* in * return filterNotTo(ArrayList\n* in * List\n\n* in * Returns a list containing all elements that are not `null`.\n* in * in * @sample samples.collections.Collections.Filtering.filterNull\n* in * in * An public inline fun <T : Any> Iterable\n* in * filterNull(): List\n\n* in * return filterNullTo(ArrayList\n* in * Append all elements that are not `null` to
the given [destination].\n* in * in * @sample samples.collections.Collections.Filtering.filterTo\n* in * in * An public inline fun <T, C : MutableCollection\n* in * C> Iterable\n* in * filterTo(destina\n* in * in * Append all elements that are not matching
the given [predicate] to the given [destination].\n* in * in * @sample samples.collections.Collections.Filtering.filter\n* in * in * An public inline fun <T, C : MutableCollection\n* in * C> Iterable\n* in * filter\n* in * in * Returns a list containing all elements matching
the given [predicate].\n* in * in * @sample samples.collections.Collections.Filtering.filter\n* in * in * Returns a list containing all elements matching
the given [predicate].\n* in * in * @sample samples.collections.Collections.Filtering.filter\n* in * in * Returns a list containing all elements matching
the given [predicate].\n* in * in * @sample samples.collections.Collections.Filtering.filter\n* in * in * Returns a list containing all elements matching
the given [predicate].\n* in * in * @sample samples.collections.Collections.Filtering.filter\n* in * in * Returns a list containing all elements matching
the given [predicate].\n* in * in * @sample samples.collections.Collections.Filtering.filter\n* in * in * Returns a list containing all elements matching
the given [predicate].\n* in * in * @sample samples.collections.Collections.Filtering.filter
@SinceKotlin("1.3")
public fun <T> MutableList<T>.shuffle(random: Random): Unit {
    for (i in lastIndex downTo 1) {
        val j = random.nextInt(i + 1)
        this[j] = this[i]
        this[i] = this[j]
    }
}

/**
 * Sorts elements in the list in-place according to natural sort order of the value returned by specified [selector] function.
 * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.
 * */
public fun <T : Comparable<T>> MutableList<T>.sortDescending(): Unit {
    sortWith(reverseOrder())
}

/**
 * Returns a list of all elements sorted according to their natural sort order.
 * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.
 * */
public fun <T : Comparable<T>> Iterable<T>.sortedDescending(): List<T> {
    return sortedWith(reverseOrder())
}

/**
 * Returns a list of all elements sorted according to the specified [comparator].
 * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.
 * */
public fun <T> Iterable<T>.sortedWith(comparator: Comparator<in T>): List<T> {
    if (this is Collection) {
        if (size <= 1) return this.toList()
        return toTypedArray<Any?>().asList()
    }
    return toMutableList().apply { sortWith(comparator) }
}

/**
 * Returns a list of all elements sorted according to the specified [selector] function.
 * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.
 * */
public fun <T : Comparable<T>> MutableList<T>.sortByDescending(crossinline selector: (T) -> R?): Unit {
    if (size > 1)
        sortWith(compareByDescending(selector))
}

/**
 * Returns a list of all elements sorted according to natural sort order of the value returned by specified [selector] function.
 * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.
 * */
public fun <T : Comparable<T>> MutableList<T>.sortBy(crossinline selector: (T) -> R?): Unit {
    if (size > 1)
        sortWith(compareBy(selector))
}

/**
 * Returns a list of all elements sorted ascending according to natural sort order.
 * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.
 * */
public fun <T : Comparable<T>> Iterable<T>.sortedBy(crossinline selector: (T) -> R?): List<T> {
    return sortedWith(compareBy(selector))
}

/**
 * Returns a list of all elements sorted descending according to natural sort order.
 * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.
 * */
public fun <T : Comparable<T>> Iterable<T>.sortedByDescending(crossinline selector: (T) -> R?): List<T> {
    return sortedWith(compareByDescending(selector))
}

/**
 * Returns a list of all elements sorted according to their natural sort order.
 * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.
 * */
public fun <T : Comparable<T>> Collection<T>.sorted(): List<T> {
    if (this is Collection) {
        if (size <= 1) return this.toList()
        return toTypedArray<Comparable<T>>().asList()
    }
    return toMutableList().apply { sort() }
}

/**
 * Returns a list of all elements sorted according to the specified [comparator].
 * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.
 * */
public fun <T> Collection<T>.sortedWith(comparator: Comparator<in T>): List<T> {
    if (this is Collection) {
        if (size <= 1) return this.toList()
        return toTypedArray<Any?>().asList()
    }
    return toMutableList().apply { sortWith(comparator) }
}

/**
 * Returns a list of all elements sorted according to their natural sort order.
 * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.
 * */
public fun <T : Comparable<T>> Collection<T>.sortedDescending(): List<T> {
    return sortedWith(reverseOrder())
}

/**
 * Returns a list of all elements sorted according to the specified [comparator].
 * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.
 * */
public fun <T> Collection<T>.sortedWith(comparator: Comparator<in T>): List<T> {
    if (this is Collection) {
        if (size <= 1) return this.toList()
        return toTypedArray<Comparable<T>>().asList()
    }
    return toMutableList().apply { sortWith(comparator) }
}

/**
 * Returns a list of all elements sorted according to natural sort order.
 * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.
 * */
public fun <T : Comparable<T>> Collection<T>.sortDescending(): Unit {
    sortWith(reverseOrder())
}

/**
 * Returns a list of all elements sorted according to the specified [comparator].
 * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.
 * */
public fun <T> Collection<T>.sortedWith(comparator: Comparator<in T>): List<T> {
    if (this is Collection) {
        if (size <= 1) return this.toList()
        return toTypedArray<Comparable<T>>().asList()
    }
    return toMutableList().apply { sortWith(comparator) }
}

/**
 * Returns a list of all elements sorted according to their natural sort order.
 * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.
 * */
public fun <T : Comparable<T>> Collection<T>.sorted(): List<T> {
    if (this is Collection) {
        if (size <= 1) return this.toList()
        return toTypedArray<Comparable<T>>().asList()
    }
    return toMutableList().apply { sort() }
}

/**
 * Returns a list of all elements sorted according to the specified [comparator].
 * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.
 * */
public fun <T> Collection<T>.sortedWith(comparator: Comparator<in T>): List<T> {
    if (this is Collection) {
        if (size <= 1) return this.toList()
        return toTypedArray<Comparable<T>>().asList()
    }
    return toMutableList().apply { sortWith(comparator) }
}

/**
 * Returns an array of Boolean containing all of the elements of this collection.
 * The array elements are in the same order as the collection.
 * */
public fun Collection<Boolean>.toBooleanArray(): BooleanArray {
    val result = BooleanArray(size)
    var index = 0
    for (element in this)
        result[index++] = element
    return result
}

/**
 * Returns an array of Byte containing all of the elements of this collection.
 * The array elements are in the same order as the collection.
 * */
public fun Collection<Byte>.toByteArray(): ByteArray {
    val result = ByteArray(size)
    var index = 0
    for (element in this)
        result[index++] = element
    return result
}

/**
 * Returns an array of Char containing all of the elements of this collection.
 * The array elements are in the same order as the collection.
 * */
public fun Collection<Char>.toCharArray(): CharArray {
    val result = CharArray(size)
    var index = 0
    for (element in this)
        result[index++] = element
    return result
}

/**
 * Returns an array of Double containing all of the elements of this collection.
 * The array elements are in the same order as the collection.
 * */
public fun Collection<Double>.toDoubleArray(): DoubleArray {
    val result = DoubleArray(size)
    var index = 0
    for (element in this)
        result[index++] = element
    return result
}

/**
 * Returns an array of Float containing all of the elements of this collection.
 * The array elements are in the same order as the collection.
 * */
public fun Collection<Float>.toFloatArray(): FloatArray {
    val result = FloatArray(size)
    var index = 0
    for (element in this)
        result[index++] = element
    return result
}

/**
 * Returns an array of Int containing all of the elements of this collection.
 * The array elements are in the same order as the collection.
 * */
public fun Collection<Int>.toIntArray(): IntArray {
    val result = IntArray(size)
    var index = 0
    for (element in this)
        result[index++] = element
    return result
}

/**
 * Returns an array of Long containing all of the elements of this collection.
 * The array elements are in the same order as the collection.
 * */
public fun Collection<Long>.toLongArray(): LongArray {
    val result = LongArray(size)
    var index = 0
    for (element in this)
        result[index++] = element
    return result
}

/**
 * Returns an array of Short containing all of the elements of this collection.
 * The array elements are in the same order as the collection.
 * */
public fun Collection<Short>.toShortArray(): ShortArray {
    val result = ShortArray(size)
    var index = 0
    for (element in this)
        result[index++] = element
    return result
}

/**
 * Returns an array of String containing all of the elements of this collection.
 * The array elements are in the same order as the collection.
 * */
public fun Collection<String>.toStringArray(): StringArray {
    val result = StringArray(size)
    var index = 0
    for (element in this)
        result[index++] = element
    return result
}

/**
 * Returns an array of Unit containing all of the elements of this collection.
 * The array elements are in the same order as the collection.
 * */
public fun Collection<Unit>.toUnitArray(): UnitArray {
    val result = UnitArray(size)
    var index = 0
    for (element in this)
        result[index++] = Unit
    return result
}
```kotlin
result[index++] = element
    return result
}

/**
 * Returns an array of Long containing all of the elements of this collection.
 */
@SinceKotlin("1.3")
public fun Collection<Long>.toLongArray(): LongArray {
    val result = LongArray(size)
    var index = 0
    for (element in this)
        result[index++] = element
    return result
}

/**
 * Returns an array of Short containing all of the elements of this collection.
 */
public fun Collection<Short>.toShortArray(): ShortArray {
    val result = ShortArray(size)
    var index = 0
    for (element in this)
        result[index++] = element
    return result
}

/**
 * Returns a [Map] containing key-value pairs provided by [transform] function
 * applied to elements of the given collection.
 */
public inline fun <T, K, V>
    Iterable<T>.associate(transform: (T) -> Pair<K, V>): Map<K, V> {
    val capacity = mapCapacity(collectionSizeOrDefault(10)).coerceAtLeast(16)
    return associateTo(LinkedHashMap<K, V>(capacity), transform)
}

/**
 * Returns a [Map] containing the elements from the given collection indexed
 * by the key returned from [keySelector] function applied to each element.
 */
public inline fun <T, K, M : MutableMap<K, in T>>
    Iterable<T>.associateByTo(destination: M, keySelector: (T) -> K): M {
    for (element in this) {
        destination.put(keySelector(element), element)
    }
    return destination
}

/**
 * Returns a [Map] where keys are elements from the given collection and values are
 * produced by the [valueSelector] function applied to each element.
 */
public inline fun <T, K, V, M : MutableMap<K, in V>>
    Iterable<T>.associateWith(valueSelector: (T) -> V): Map<K, V> {
    for (element in this) {
        destination.put(element, valueSelector(element))
    }
    return destination
}
```

---

```kotlin
result[index++] = element
    return result
}
```

---

```kotlin
val result = LongArray(size)
var index = 0
for (element in this)
    result[index++] = element
return result
}
```

---

```kotlin
val result = ShortArray(size)
var index = 0
for (element in this)
    result[index++] = element
return result
}
```

---

```kotlin
val capacity = mapCapacity(collectionSizeOrDefault(10)).coerceAtLeast(16)
return associateTo(LinkedHashMap<K, V>(capacity), transform)
```

---

```kotlin
val capacity = mapCapacity(collectionSizeOrDefault(10)).coerceAtLeast(16)
return associateByTo(LinkedHashMap<K, T>(capacity), keySelector)
```

---

```kotlin
val capacity = mapCapacity(collectionSizeOrDefault(10)).coerceAtLeast(16)
return associateByTo(LinkedHashMap<K, T>(capacity), keySelector, valueTransform)
```

---

```kotlin
val capacity = mapCapacity(collectionSizeOrDefault(10)).coerceAtLeast(16)
return associateTo(LinkedHashMap<K, V>(capacity), transform)
```

---

```kotlin
val capacity = mapCapacity(collectionSizeOrDefault(10)).coerceAtLeast(16)
return associateByTo(LinkedHashMap<K, T>(capacity), keySelector)
```

---

```kotlin
val capacity = mapCapacity(collectionSizeOrDefault(10)).coerceAtLeast(16)
return associateByTo(LinkedHashMap<K, T>(capacity), keySelector)
```

---

```kotlin
val result = LongArray(size)
var index = 0
for (element in this)
    result[index++] = element
return result
```

---

```kotlin
val result = ShortArray(size)
var index = 0
for (element in this)
    result[index++] = element
return result
```

---

```kotlin
val capacity = mapCapacity(collectionSizeOrDefault(10)).coerceAtLeast(16)
return associateTo(LinkedHashMap<K, V>(capacity), transform)
```
Iterable<K>.associateWith(valueSelector: (K) -> V): Map<K, V> {
    val result = Linked_hashMap<K, V>(mapCapacity(collectionSizeOrDefault(10)).coerceAtLeast(16))
    return associateWithTo(result, valueSelector)
}

Iterable<K>.associateWithTo(destination: M, valueSelector: (K) -> V): M {
    for (element in this) {
        destination.put(element, valueSelector(element))
    }
    return destination
}

Iterable<K>.toCollection(destination: C): C {
    for (item in this) {
        destination.add(item)
    }
    return destination
}

Iterable<K>.toHashSet(): HashSet<T> {
    return toCollection(HashSet<T>(mapCapacity(size)))
}

Iterable<T>.flatMap(transform: (T) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}

Iterable<T>.flatMapIndexed(transform: (index: Int, T) -> Iterable<R>): List<R> {
    return flatMapIndexedTo(ArrayList<R>(), transform)
}

Iterables.flatMapTo(Iterable<T>, Function1<T, R>): List<R> {
    return flatMapTo(ArrayList<R>(), Function1(this))
}

Iterables.flatMapIndexedTo(Iterable<T>, Function1(index: Int, T, R)): List<R> {
    return flatMapIndexedTo(ArrayList<R>(), Function1(index, this))
}

@SinceKotlin("1.3")
public inline fun <K, V, M : MutableMap<K, V>>
    Iterable<K>.associateWithTo(destination: M, valueSelector: (K) -> V): M {
    val result = LinkedHashMap<K, V>(mapCapacity(collectionSizeOrDefault(10)).coerceAtLeast(16))
    return associateWithTo(result, valueSelector)
}

/**
 * Populates and returns the [destination] mutable map with key-value pairs for each element of the given collection.
 * If any two elements are equal, the last one overwrites the former value in the map.
 * @sample samples.collections.Transformations.associateWithTo
 */
@SinceKotlin("1.3")
public inline fun <K, V, M : MutableMap<K, V>>
    Iterable<K>.associateWithTo(destination: M, valueSelector: (K) -> V): M {
    for (element in this) {
        destination.put(element, valueSelector(element))
    }
    return destination
}

/**
 * Appends all elements to
 * @sample samples.collections.Transformations.toCollection
 */
public fun <T, C : MutableCollection<in T>>
    Iterable<T>.toCollection(destination: C): C {
    for (item in this) {
        destination.add(item)
    }
    return destination
}

/**
 * Returns a new [HashSet] of all elements.
 * @sample samples.collections.Transformations.toHashSet
 */
public fun <T> Iterable<T>.toHashSet(): HashSet<T> {
    return toCollection(HashSet<T>(mapCapacity(size)))
}

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original collection.
 * @sample samples.collections.Transformations.flatMap
 */
public inline fun <T, R> Iterable<T>.flatMap(transform: (T) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each element
 * and its index in the original collection.
 * @sample samples.collections.Transformations.flatMapIndexed
 */
public inline fun <T, R> Iterable<T>.flatMapIndexed(transform: (index: Int, T) -> Iterable<R>): List<R> {
    return flatMapIndexedTo(ArrayList<R>(), transform)
}
public inline fun <T, R, C : MutableCollection<in R>> Iterable<T>.flatMapIndexedTo(destination: C, transform: (Int, T) -> Iterable<R>): C {
    var index = 0
    for (element in this) {
        val list = transform(checkIndexOverflow(index++), element)
        destination.addAll(list)
    }
    return destination
}

/**
 * Appends all elements yielded from results of [transform] function being invoked on each element in the original collection, to the given [destination].
 */
public inline fun <T, R, C : MutableCollection<in R>> Iterable<T>.flatMapIndexedSequenceTo(destination: C, transform: (index: Int, T) -> Sequence<R>): C {
    var index = 0
    for (element in this) {
        val list = transform(checkIndexOverflow(index++), element)
        destination.addAll(list)
    }
    return destination
}

/**
 * Groups elements of the original collection by the key returned by the given [keySelector] function and returns a map where each group key is associated with a list of corresponding elements.
 */
public inline fun <T, K> Iterable<T>.groupBy(keySelector: (T) -> K): Map<K, List<T>> {
    return groupByTo(LinkedHashMap<K, MutableList<T>>(), keySelector)
}

/**
 * Groups values returned by the [valueTransform] function applied to each element of the original collection
 * by the key returned by the given [keySelector] function and puts to the [destination] map each group key associated with a list of corresponding values.
 */
public inline fun <T, K, V, M : MutableMap<in K, MutableList<V>>> Iterable<T>.groupByTo(destination: M, keySelector: (T) -> K, valueTransform: (T) -> V): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<T>() }
        list.add(element)
    }
    return destination
}
destination.getOrPut(key) { ArrayList<V>() }
        list.add(valueTransform(element))
    }
    return
}

/**
 * Creates a [Grouping] source from a collection to be used later with one of group-and-fold
 * operations\n * using the specified [keySelector] function to extract a key from each element.\n * \n * @param samples\n * collections.Collections.Grouping\n * groupingByEachCount\n * *\n * \n * @sample
 */
public inline fun <T, K>
    Iterable<T>.groupingBy(crossinline keySelector: (T) -> K): Grouping<T, K> \n    { \n        Grouping<T, K> { \n            return override fun sourceIterator(): Iterator<T> = this@groupingBy.iterator()
            override fun keyOf(element:
                T): K = keySelector(element)
        }
    }

/**
 * Returns a list containing the results of applying the given [transform] function to each
 * element in the original collection.\n * \n * @param \n * samples.collections.Collections.Transformations.map\n * *\n * @public inline fun <T, R> \n * Iterable<T>.map(transform: \n * (T) -> R): List<R> \n * \n * @sample
 */
public inline fun <T, R> \n    Iterable<T>.map(transform: (T) -> R): List<R> \n    { \n        return mapTo(ArrayList<R>(collectionSizeOrDefault(10)), transform)\n    }

/**
 * Returns a list containing the results of applying the given [transform] function to each
 * element in the original collection.\n * \n * @param \n * @sample
 */
public inline fun <T, R : Any> \n    Iterable<T>.mapNotNull(transform: (T) -> R?): List<R> \n    { \n        return mapNotNullTo(ArrayList<R>(), transform)\n    }

/**
 * Applies the given [transform] function to each element of the original collection
 * and appends the results to the given \n * collection.\n * \n * @param \n * @sample
 */
public inline fun <T, R, C : MutableCollection<in R>> \n    Iterable<T>.mapTo(destination: C, transform: (T) -> R): C \n    { \n        for (item in this)
            destination.add(transform(item))
    }

/**
 * Returns a list containing only distinct elements from the given collection.\n * \n * Among equal elements of the given collection, only the first one will be present in the
 * resulting list.\n * \n * The elements in the resulting list are in the same order as they were in the source
 * collection.\n * \n * @sample
 */
public fun <T> \n    Iterable<T>.distinct():
List<T> { return this.toMutableSet().toList() } * Returns a list containing only elements from the given collection

Iterable<T>.distinctBy(selector: (T) -> K): List<T> { val set = HashSet<K>() val list = ArrayList<T>() for (e in this) { val key = selector(e) if (set.add(key)) list.add(e) } return list } * Returns a set containing all elements that are contained by both this collection and the specified collection. The returned set preserves the element iteration order of the original collection.

Iterable<T>.intersect(other: Iterable<T>): Set<T> { val set = this.toMutableSet() set.retainAll(other) return set } * Returns a set containing all elements that are contained in at least one of these collections use [union].

Iterable<T>.subtract(other: Iterable<T>): Set<T> { val set = this.toMutableSet() set.removeAll(other) return set } * Returns a set containing all elements that are contained by this collection and not contained by the specified collection.

Iterable<T>.toMutableSet(): MutableSet<T> { return when (this) { is Collection<T> -> LinkedHashSet(this) else -> toCollection(LinkedHashSet<T>()) } } * Returns a new [MutableSet] containing all distinct elements from the given collection.

Iterable<T>.union(other: Iterable<T>): Set<T> { val set = this.toMutableSet() set.addAll(other) return set } * Returns a set containing all distinct elements from both collections.

Iterable<T>.any(): Boolean { if (this is Collection) return !isEmpty() return iterator().hasNext() } * Returns `true` if at least one element matches the given [predicate].

Iterable<T>.any(predicate: (T) -> Boolean): Boolean { if (this is Collection && isEmpty()) return false

var count = 0

for (element in this) if (predicate(element)) checkCountOverflow(++count)

return count } * Returns the number of elements matching the given [predicate].

Iterable<T>.fold(initial: R, operation: (acc: R, T) -> R): R { var accumulator = initial

for (element in this) accumulator = operation(accumulator, element)

return accumulator } * Accumulates value starting with [initial] value and applying [operation] from left to right. If [operation] takes current accumulator value and an element, and calculates the next accumulator value, use [foldIndexed].

Iterable<T>.foldIndexed(initial: R, operation: (index: Int, acc: R, T) -> R): R { var accumulator = initial

for (element in this) accumulator = operation(accumulator, element) return accumulator }
index = 0
var accumulator = initialValue
for (element in this) accumulator =
operation(checkIndexOverflow(index++), accumulator, element)
return accumulator

/*
 * Accumulates value starting with [initial] value and applying [operation] from left to right to each element and current
accumulator value. */

/**
 * Accumulates value starting with [initial] value and applying [operation] from right to left to each element and current
accumulator value. */

/**
 * Performs the given [action] on each element.
 */
@kotlin.internal.HidesMembers
public inline fun <T> Iterable<T>.forEach(action: (T) -> Unit): Unit {
    for (element in this) action(element)
}

/**
 * Performs the given [action] on each element, providing sequential
 * index with the element.
 * @param [action] function that takes the index of an element and the element itself
 * and performs the action on the element.
 */
public inline fun <T> Iterable<T>.forEachIndexed(action: (index: Int, T) -> Unit): Unit {
    var index = 0
    for (item in this) action(checkIndexOverflow(index++), item)
}

@Deprecated("Use maxOrNull instead.", ReplaceWith("this.maxOrNull()"))
@DeprecatedSinceKotlin(warningSince = "1.4")
@SinceKotlin("1.1")
public fun Iterable<Double>.max(): Double? {
    return maxOrNull()
}

@Deprecated("Use maxOrNull instead.", ReplaceWith("this.maxOrNull()"))
@DeprecatedSinceKotlin(warningSince = "1.4")
@SinceKotlin("1.1")
public fun Iterable<Float>.max(): Float? {
    return maxOrNull()
}

@Deprecated("Use maxOrNull instead.", ReplaceWith("this.maxOrNull()"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun <T : Comparable<T>> Iterable<T>.max(): T? {
    return maxOrNull()
}

@Deprecated("Use maxByOrNull instead.", ReplaceWith("this.maxByOrNull(selector)"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public inline fun <T, R : Comparable<R>> Iterable<T>.maxBy(selector: (T) -> R): T? {
    return maxByOrNull(selector)
}
produced by [selector] function is 'NaN', the returned result is 'NaN'.

*\@SinceKotlin("1.4\")\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\@kotlin.internal.InlineOnly\@public inline fun \(<T\> \@Iterable\<T\>.maxOf(selector: \(T\) -> Float): Float \{\n  val iterator = iterator()\n  if (!iterator.hasNext()) throw NoSuchElementException()\n  var maxValue = selector(iterator.next())\n  while (iterator.hasNext()) {\n    val v = selector(iterator.next())\n    maxValue = maxOf(maxValue, v)\n  }\n  return maxValue\n\} \n* \n* @throws NoSuchElementException if the collection is empty.\n
*\@SinceKotlin("1.4\")\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\@kotlin.internal.InlineOnly\@public inline fun \(<T, R : Comparable\<R>> \@Iterable\<T\>.maxOfNull(selector: \(T\) -> R): R \{\n  val iterator = iterator()\n  if (!iterator.hasNext()) throw NoSuchElementException()\n  var maxValue = selector(iterator.next())\n  while (iterator.hasNext()) {\n    val v = selector(iterator.next())\n    if (maxValue < v) {\n      maxValue = v\n    }\n  }\n  return maxValue\n\} \n* \n* Returns the largest value among all values produced by [selector] function \n* applied to each element in the collection or 'null' if there are no elements.\n
*\@SinceKotlin("1.4\")\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\@kotlin.internal.InlineOnly\@public inline fun \(<T\> \@Iterable\<T\>.maxOfWith\(\) \(\text{comparator: Comparator\langle\text{in R}\rangle, selector: (T) -> R: R}\): R \{\n  val iterator = iterator()\n  if (!iterator.hasNext()) throw NoSuchElementException()\n  var maxValue = selector(iterator.next())\n  while (iterator.hasNext()) {\n    val v = selector(iterator.next())\n    if (comparator.compare(maxValue, v) < 0) {\n      maxValue = v\n    }\n  }\n  return maxValue\n\} \n* \n* Returns the largest value according to the provided [comparator] \n* among all values produced by [selector] function applied to each element in the collection or 'null' if there are no elements.\n
*\@SinceKotlin("1.4\")\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\@kotlin.internal.InlineOnly\@public inline fun \(<T, R : Comparable\<R>> \@Iterable\<T\>.maxOfWithNull\(\) \(\text{comparator: Comparator\langle\text{in R}\rangle, selector: (T) -> R: null}\): R? \{\n  val iterator = iterator()\n  if (!iterator.hasNext()) return null\n  var maxValue = selector(iterator.next())\n  while (iterator.hasNext()) {\n    val v = selector(iterator.next())\n    if (comparator.compare(maxValue, v) < 0) {\n      maxValue = v\n    }\n  }\n  return maxValue\n\} \n* \n* Returns the largest value according to the provided [comparator] \n* among all values produced by [selector] function applied to each element in the collection or 'null' if there are no elements.\n
*\@SinceKotlin("1.4\")\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\@kotlin.internal.InlineOnly\@public inline fun \(<T, R : Comparable\<R>> \@Iterable\<T\>.maxOfWithNull\(\) \(\text{comparator: Comparator\langle\text{in R}\rangle, selector: (T) -> R: null}\): R? \{\n  val iterator = iterator()\n  if (!iterator.hasNext()) return null\n  var maxValue = selector(iterator.next())\n  while (iterator.hasNext()) {\n    val v = selector(iterator.next())\n    if (comparator.compare(maxValue, v) < 0) {\n      maxValue = v\n    }\n  }\n  return maxValue\n\} \n* \n* Returns the largest value according to the provided [comparator] \n* among all values produced by [selector] function applied to each element in the collection or 'null' if there are no elements.\n
*\@SinceKotlin("1.4\")\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\@kotlin.internal.InlineOnly\@public inline fun \(<T, R : Comparable\<R>> \@Iterable\<T\>.maxOfWithNull\(\) \(\text{comparator: Comparator\langle\text{in R}\rangle, selector: (T) -> R: null}\): R? \{\n  val iterator = iterator()\n  if (!iterator.hasNext()) return null\n  var maxValue = selector(iterator.next())\n  while (iterator.hasNext()) {\n    val v = selector(iterator.next())\n    if (comparator.compare(maxValue, v) < 0) {\n      maxValue = v\n    }\n  }\n  return maxValue\n\} \n* \n* Returns the largest value according to the provided [comparator] \n* among all values produced by [selector] function applied to each element in the collection or 'null' if there are no elements.
Iterable<T>.maxOfWithOrNull(comparator: Comparator<in R>, selector: (T) -> R): R? { 
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var maxElement = iterator.next()
    while (iterator.hasNext()) {
        val candidate = iterator.next()
        if (comparator.compare(maxElement, candidate) < 0) {
            maxElement = candidate
        }
    }
    return maxElement
}

/**
 * Returns the largest element or `null` if there are no elements.
 * @since 1.4
 */
@SinceKotlin("1.4")
public fun Iterable<Double>.maxOrNull(): Double? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var max = iterator.next()
    while (iterator.hasNext()) {
        val e = iterator.next()
        max = maxOf(max, e)
    }
    return max
}

/**
 * Returns the largest element or `null` if there are no elements.
 * @since 1.4
 */
@SinceKotlin("1.4")
public fun Iterable<Float>.maxOrNull(): Float? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var max = iterator.next()
    while (iterator.hasNext()) {
        val e = iterator.next()
        max = maxOf(max, e)
    }
    return max
}

/**
 * Returns the largest element or `null` if there are no elements.
 * @since 1.4
 */
@SinceKotlin("1.4")
public fun <T : Comparable<T>> Iterable<T>.maxOrNull(): T? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var max = iterator.next()
    while (iterator.hasNext()) {
        val candidate = iterator.next()
        if (max < candidate) max = candidate
    }
    return max
}

@Deprecated("Use maxWithOrNull instead.",
ReplaceWith("this.maxWithOrNull(comparator)"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public inline fun <T> Iterable<T>.maxWithOrNull(comparator: Comparator<in T>): T? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var max = iterator.next()
    while (iterator.hasNext()) {
        val candidate = iterator.next()
        if (comparator.compare(max, candidate) < 0) {
            max = candidate
        }
    }
    return max
}

@Deprecated("Use minOrNull instead.",
ReplaceWith("this.minOrNull()"))
@DeprecatedSinceKotlin(warningSince = "1.4")
@SinceKotlin("1.1")
public fun Iterable<Double>.min(): Double? {
    return minOrNull()
}

@Deprecated("Use minOrNull instead.",
ReplaceWith("this.minOrNull()"))
@DeprecatedSinceKotlin(warningSince = "1.4")
@SinceKotlin("1.1")
public fun Iterable<Float>.min(): Float? {
    return minOrNull()
}

@Deprecated("Use minOrNull instead.",
ReplaceWith("this.minOrNull()"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun <T : Comparable<T>> Iterable<T>.min(): T? {
    return minOrNull()
}

@Deprecated("Use minByOrNull instead.",
ReplaceWith("this.minByOrNull(selector)"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public inline fun <T, R : Comparable<R>> Iterable<T>.minByOrNull(selector: (T) -> R): T? {
    val iterator = iterator()
    if (!iterator.hasNext()) throw NoSuchElementException()
    var minValue = selector(iterator.next())
    while (iterator.hasNext()) {
        val v = selector(iterator.next())
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}
produced by [selector] function is \'NaN\', the returned result is \'NaN\'. 

```kotlin
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType@kotlin.internal.InlineOnly
@public inline fun <T> Iterable<T>.minOfOrNull(selector: (T) -> Double): Double? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var minValue = selector(iterator.next())
    while (iterator.hasNext()) {
        val v = selector(iterator.next())
        if (minValue > v) { minValue = v }
    }
    return minValue
}
```

```
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType@kotlin.internal.InlineOnly
@public inline fun <T, R : Comparable<R>>
Iterable<T>.minOfOrNull(selector: (T) -> R): R {
    val iterator = iterator()
    if (!iterator.hasNext()) throw NoSuchElementException()
    var minValue = selector(iterator.next())
    while (iterator.hasNext()) {
        val v = selector(iterator.next())
        if (minValue > v) { minValue = v }
    }
    return minValue
}
```

```
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType@kotlin.internal.InlineOnly
@public inline fun <T> Iterable<T>.minOfOrNull(selector: (T) -> Float): Float? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var minValue = selector(iterator.next())
    while (iterator.hasNext()) {
        val v = selector(iterator.next())
        if (minValue > v) { minValue = v }
    }
    return minValue
}
```

```
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType@kotlin.internal.InlineOnly
@public inline fun <T, R : Comparable<R>>
Iterable<T>.minOfOrNull(selector: (T) -> R): R? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var minValue = selector(iterator.next())
    while (iterator.hasNext()) {
        val v = selector(iterator.next())
        if (comparator.compare(minValue, v) > 0) { minValue = v }
    }
    return minValue
}
```
iterator() if (!iterator.hasNext()) return null

@SinceKotlin("1.4")
public fun <T> Iterable<T>.minOrNull(): T? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var min = iterator.next()
    while (iterator.hasNext()) {
        val e = iterator.next()
        if (min > e) min = e
    }
    return min
}

/**
 * Returns the smallest element or `null` if there are no elements.
 * 
 * If any of elements is `NaN` returns `NaN`.
 */
@SinceKotlin("1.4")
public fun <T : Comparable<T>> Iterable<T>.minOrNull(comparator: Comparator<T>): T? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var minValue = iterator.next()
    while (iterator.hasNext()) {
        val v = iterator.next()
        if (comparator.compare(minValue, v) > 0) minValue = v
    }
    return minValue
}

/**
 * Returns the first element having the smallest value according to the provided [comparator] or `null` if there are no elements.
 * 
 * If any of elements is `NaN` returns `NaN`.
 */
@SinceKotlin("1.4")
public fun <T> Iterable<T>.reduceIndexed(operation: (index: Int, acc: T, T) -> T): T? {
    val iterator = this.iterator()
    if (!iterator.hasNext()) throw UnsupportedOperationException("Empty collection can't be reduced.")
    var accumulator = iterator.next()
    var index = 0
    while (iterator.hasNext()) {
        accumulator = operation(index, accumulator, iterator.next())
        index++
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right on
 * the current accumulator value and each element with its index in the original collection.
 * 
 * @throws SupportsUnsupportedOperationException
 */
@SinceKotlin("1.4")
public fun <S, T : S> Iterable<T>.reduceIndexed(operation: (index: Int, T) -> S): S? {
    val iterator = this.iterator()
    if (!iterator.hasNext()) throw UnsupportedOperationException("Empty collection can't be reduced.")
    var accumulator: S = null
    var index = 0
    while (iterator.hasNext()) {
        accumulator = operation(index, accumulator, iterator.next())
        index++
    }
    return accumulator
}

/**
 * Returns the smallest element or `null` if there are no elements.
 * 
 * If any of elements is `NaN` returns `NaN`.
 */
@SinceKotlin("1.4")
public fun <T> Iterable<T>.reduce(operation: (acc: T, T) -> T): T? {
    val iterator = this.iterator()
    if (!iterator.hasNext()) return null
    var min = iterator.next()
    while (iterator.hasNext()) {
        val e = iterator.next()
        if (min > e) min = e
    }
    return min
}

/**
 * Returns true if the collection has no elements.
 * 
 * @sample samples.collections.Collections.Aggregates.none
 */
public fun <T> Iterable<T>.none(): Boolean {
    if (this is Collection) return isEmpty()
    return !iterator().hasNext()
}

/**
 * Returns true if no elements match the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.noneWithPredicate
 */
public inline fun <T> Iterable<T>.none(predicate: (T) -> Boolean): Boolean {
    if (this is Collection && isEmpty()) return true
    for (element in this) if (predicate(element)) return false
    return true
}

/**
 * Performs the given [operation] on each element and returns the collection itself afterwards.
 * 
 * @param [operation] function that takes current accumulator value and an element
 * and returns the next accumulator value.
 * 
 * @throws SupportsUnsupportedOperationException
 */
@SinceKotlin("1.4")
public inline fun <S, T : S> Iterable<T>.reduceIndexed(operation: (Int, T) -> S): S? {
    val iterator = this.iterator()
    if (!iterator.hasNext()) throw UnsupportedOperationException("Empty collection can't be reduced.")
    var accumulator: S = null
    var index = 0
    while (iterator.hasNext()) {
        accumulator = operation(index, accumulator, iterator.next())
        index++
    }
    return accumulator
}

/**
 * Performs the given [operation] on each element and returns the collection itself afterwards.
 * 
 * @throws SupportsUnsupportedOperationException
 */
@SinceKotlin("1.4")
public inline fun <S, T : S> Iterable<T>.reduce(operation: (S, T) -> S): S? {
    val iterator = this.iterator()
    if (!iterator.hasNext()) throw UnsupportedOperationException("Empty collection can't be reduced.")
    var accumulator: S = null
    while (iterator.hasNext()) {
        accumulator = operation(accumulator, iterator.next())
    }
    return accumulator
}

/**
 * Returns the smallest element or `null` if there are no elements.
 * 
 * If any of elements is `NaN` returns `NaN`.
 */
@SinceKotlin("1.4")
public fun <T> Iterable<T>.minOrNull(): T? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var minValue = iterator.next()
    while (iterator.hasNext()) {
        val v = iterator.next()
        if (comparator.compare(minValue, v) > 0) minValue = v
    }
    return minValue
}

/**
 * Returns the first element having the smallest value according to the provided [comparator] or `null` if there are no elements.
 * 
 * If any of elements is `NaN` returns `NaN`.
 */
@SinceKotlin("1.4")
public fun <T : Comparable<T>> Iterable<T>.minOrNull(comparator: Comparator<T>): T? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var minValue = iterator.next()
    while (iterator.hasNext()) {
        val v = iterator.next()
        if (comparator.compare(minValue, v) > 0) minValue = v
    }
    return minValue
}

/**
 * Performs the given [operation] on each element and returns the collection itself afterwards.
 * 
 * @throws SupportsUnsupportedOperationException
 */
@SinceKotlin("1.4")
public inline fun <S, T : S> Iterable<T>.reduceIndexed(operation: (Int, T) -> S): S? {
    val iterator = this.iterator()
    if (!iterator.hasNext()) throw UnsupportedOperationException("Empty collection can't be reduced.")
    var accumulator: S = null
    var index = 0
    while (iterator.hasNext()) {
        accumulator = operation(index, accumulator, iterator.next())
        index++
    }
    return accumulator
}

/**
 * Performs the given [operation] on each element and returns the collection itself afterwards.
 * 
 * @throws SupportsUnsupportedOperationException
 */
@SinceKotlin("1.4")
public inline fun <S, T : S> Iterable<T>.reduce(operation: (S, T) -> S): S? {
    val iterator = this.iterator()
    if (!iterator.hasNext()) throw UnsupportedOperationException("Empty collection can't be reduced.")
    var accumulator: S = null
    while (iterator.hasNext()) {
        accumulator = operation(accumulator, iterator.next())
    }
    return accumulator
}
iterator = this.iterator()
    if (!iterator.hasNext()) throw UnsupportedOperationException("Empty collection can't be reduced.")
    var index = 1
    var accumulator: S = iterator.next()
    while (iterator.hasNext()) {
        accumulator = operation(checkIndexOverflow(index++), accumulator, iterator.next())
        index = index + 1
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying \[operation\] from left to right \n * to current accumulator value and each element with its index in the original collection. \n * \n * @param \[operation\] function that takes the index of an element, current accumulator value \n * and the element itself; \n * and calculates the next accumulator value. \n * @sample samples.collections.Collections.Aggregates.reduceOrNonNull
 */
@SinceKotlin("1.4")
public inline fun <S, T : S> Iterable<T>.reduceOrNonNull(operation: (index: Int, acc: S, T) -> S): S? {
    val iterator = this.iterator()
    if (!iterator.hasNext()) return null
    var index = 1
    var accumulator: S = iterator.next()
    while (iterator.hasNext()) {
        accumulator = operation(checkIndexOverflow(index++), accumulator, iterator.next())
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying \[operation\] from left to right \n * to each element and applying \[operation\] from right to left \n * to each element and current accumulator value. \n * \n * @param \[operation\] function that takes current accumulator value and an element; \n * and calculates \n * the next accumulator value. \n * @sample samples.collections.Collections.Aggregates.reduceRight
 */
public inline fun <S, T : S> List<T>.reduceRight(operation: (T, acc: S) -> S): S {
    val iterator = listIterator(size)
    if (!iterator.hasPrevious())
        throw UnsupportedOperationException("Empty list can't be reduced.")
    var accumulator: S = iterator.previous()
    while (iterator.hasPrevious()) {
        accumulator = operation(iterator.previous(), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying \[operation\] from left to right \n * to each element and \n * \n * @param \[operation\] function that takes the index of an element, \n * current accumulator value \n * and calculates \n * the next accumulator value. \n * @sample samples.collections.Collections.Aggregates.reduceRightIndexed
 */
@SinceKotlin("1.4")
    val iterator = listIterator(size)
    if (!iterator.hasPrevious())
        return null
    var accumulator: S = iterator.previous()
    while (iterator.hasPrevious()) {
        val index = iterator.previousIndex()
        accumulator = operation(index, iterator.previous(), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with the last element and applying \[operation\] from right to left \n * to each element and current accumulator value. \n * \n * @param \[operation\] function that takes an element and current accumulator value, \n * and calculates \n * the next accumulator value. \n * @sample samples.collections.Collections.Aggregates.reduceRightIndexedOrNonNull
 */
public inline fun <S, T : S> List<T>.reduceRightIndexedOrNonNull(operation: (index: Int, T, acc: S) -> S?): S? {
    val iterator = listIterator(size)
    if (!iterator.hasPrevious())
        return null
    var accumulator: S = iterator.previous()
    while (iterator.hasPrevious()) {
        val index = iterator.previousIndex()
        accumulator = operation(index, iterator.previous(), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with the last element and applying \[operation\] from right to left \n * to each element and \n * \n * @param \[operation\] function that takes the index of an element, \n * current accumulator value, \n * and calculates \n * the next accumulator value. \n * @sample samples.collections.Collections.Aggregates.reduceRightOrNonNull
 */
@SinceKotlin("1.4")
public inline fun <S, T : S> List<T>.reduceRightOrNonNull(operation: (index: Int, T, acc: S) -> S?): S? {
    val iterator = listIterator(size)
    if (!iterator.hasPrevious())
        return null
    var accumulator: S = iterator.previous()
    while (iterator.hasPrevious()) {
        val index = iterator.previousIndex()
        accumulator = operation(index, iterator.previous(), accumulator)
    }
    return accumulator
}
Returns `null` if the list is empty.

```kotlin
@SinceKotlin("1.4")
public inline fun <S, T : S> List<T>.reduceRightOrNull(operation: (T, acc: S) -> S): S? {
    if (!iterator.hasPrevious()) return null
    var accumulator: S = iterator.previous()
    while (iterator.hasPrevious()) {
        accumulator = operation(iterator.previous(), accumulator)
    }
    return accumulator
}
```

This method returns a list containing successive accumulation values generated by applying the `operation` from left to right. The initial value is `null`, and the current accumulator value is updated by applying the `operation`. If the list is empty, `null` is returned.

```kotlin
@SinceKotlin("1.4")
public inline fun <T, R> Iterable<T>.runningFold(initial: R, operation: (acc: R, T) -> R): List<R> {
    val estimatedSize = collectionSizeOrDefault(9)
    if (estimatedSize == 0) return listOf(initial)
    val result = ArrayList<R>(estimatedSize + 1).apply { add(initial) }
    var accumulator = initial
    for (element in this) {
        accumulator = operation(accumulator, element)
        result.add(accumulator)
    }
    return result
}
```

This method returns a list containing successive accumulation values generated by applying the `operation` from left to right. The initial value and the current accumulator value are updated by applying the `operation`. If the estimated size is 0, a list containing only the initial value is returned.

```kotlin
@SinceKotlin("1.4")
public inline fun <S, T : S> Iterable<T>.runningReduce(operation: (acc: S, T) -> S): List<S> {
    val iterator = this.iterator()
    if (!iterator.hasNext()) return emptyList()
    var accumulator: S = iterator.next()
    val result = ArrayList<S>(collectionSizeOrDefault(10)).apply { add(accumulator) }
    while (iterator.hasNext()) {
        accumulator = operation(accumulator, iterator.next())
        result.add(accumulator)
    }
    return result
}
```

This method returns a list containing successive accumulation values generated by applying the `operation` from left to right. The initial value and the current accumulator value are updated by applying the `operation`. If the iterator is empty, an empty list is returned.
accumulator, iterator.next())
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element and current accumulator value that starts with [initial] value.
 * Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.
 * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.scan
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public inline fun <T, R> Iterable<T>.scan(initial: R, operation: (acc: R, T) -> R): List<R> {
    return runningFold(initial, operation)
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element, its index in the original collection and current accumulator value that starts with [initial] value.
 * Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.
 * @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.scan
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public inline fun <T, R> Iterable<T>.scanIndexed(initial: R, operation: (index: Int, acc: R, T) -> R): List<R> {
    return runningFoldIndexed(initial, operation)
}

@Deprecated("Use runningReduce instead.",
ReplaceWith("runningReduce(operation)"), level = DeprecationLevel.ERROR)
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public inline fun <S, T : S> Iterable<T>.scanReduce(operation: (acc: S, T) -> S): List<S> {
    return runningReduce(operation)
}

@Deprecated("Use runningReduceIndexed instead.",
ReplaceWith("runningReduceIndexed(operation)"), level = DeprecationLevel.ERROR)
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public inline fun <S, T : S> Iterable<T>.scanReduceIndexed(operation: (index: Int, acc: S, T) -> S): List<S> {
    return runningReduceIndexed(operation)
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the collection.
 * @sample samples.collections.Collections.Aggregates.sum
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@kotlin.jvm.JvmName("sumOf")
@kotlin.internal.InlineOnly
public inline fun <T> Iterable<T>.sumOf(selector: (T) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the collection.
 * @sample samples.collections.Collections.Aggregates.sum
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("sumOfDouble")
@kotlin.internal.InlineOnly
public inline fun <T> Iterable<T>.sumOfDouble(selector: (T) -> Double): Double {
    var sum: Double = 0.0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the collection.
 * @sample samples.collections.Collections.Aggregates.sum
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("sumOfLong")
@kotlin.internal.InlineOnly
public inline fun <T> Iterable<T>.sumOfLong(selector: (T) -> Long): Long {
    var sum: Long = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the collection.
 * @sample samples.collections.Collections.Aggregates.sum
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("sumOfUShort")
@kotlin.experimentalUnsignedTypes
@kotlin.internal
inline fun <T> Iterable<T>.sumOf(selector: (T) -> UInt): UInt {
    var sum: UInt = 0.toUInt()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the collection.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.jvm.JvmName("sumOfULong")
public inline fun <T> Iterable<T>.sumOf(selector: (T) -> ULong): ULong {
    var sum: ULong = 0.toULong()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns an original collection containing all the non-null elements, throwing an [IllegalArgumentException] if there are any null elements.
 */
public fun <T : Any> Iterable<T?>.requireNoNulls(): Iterable<T> {
    for (element in this) {
        if (element == null) {
            throw IllegalArgumentException("null element found in $this.")
        }
    }
    @Suppress("UNCHECKED_CAST")
    return this as Iterable<T>
}

/**
 * Returns an original collection containing all the non-null elements, throwing an [IllegalArgumentException] if there are any null elements.
 */
public fun <T : Any> List<T?>.requireNoNulls(): List<T> {
    for (element in this) {
        if (element == null) {
            throw IllegalArgumentException("null element found in $this.")
        }
    }
    @Suppress("UNCHECKED_CAST")
    return this as List<T>
}

/**
 * Splits this collection into a list of lists each not exceeding the given [size].
 * The last list in the resulting list may have fewer elements than the given [size].
 * The number of elements in each of the lists must be positive and can be greater than the number of elements in this collection.
 * @param size the number of elements to take in each list, must be positive and can be greater than the number of elements in this collection.
 * @sample samples.collections.Collections.Transformations.chunked
 */
@SinceKotlin("1.2")
public fun <T> Iterable<T>.chunked(size: Int): List<List<T>> {
    return windowed(size, size, partialWindows = true)
}

/**
 * Splits this collection into several lists each not exceeding the given [size], and applies the given [transform] function to the list passed to the [transform] function.
 * The last list may have fewer elements than the given [size].
 * The list passed to the [transform] function is ephemeral and is valid only inside that function.
 * You should not store it or allow it to escape in some way, unless you made a snapshot of it.
 * The last list may have fewer elements than the given [size].
 * The number of elements to take in each list, must be positive and can be greater than the number of elements in this collection.
 * @param size the number of elements to take in each list, must be positive and can be greater than the number of elements in this collection.
 * @param sample samples.text.Strings.chunkedTransform
 */
@SinceKotlin("1.2")
public fun <T, R> Iterable<T>.chunked(size: Int, transform: (List<T>) -> R): List<R> {
    return windowed(size, size, partialWindows = true, transform = transform)
}

/**
 * Returns a list containing all elements of the original collection that are not the first occurrence of the given [element].
 */
public operator fun <T> Iterable<T>.minus(element: T): List<T> {
    val result = ArrayList<T>(collectionSizeOrDefault(10))
    var removed = false
    return this.filterTo(result) { if (!removed && it == element) { removed = true; false } else true }
}

/**
 * Returns a list containing all elements of the original collection except the elements contained in the given [elements] sequence.
 * The elements are not contained in the given [elements] sequence.
 * @param elements the elements to remove from the list.
 * @return a list containing all elements of the original collection except the elements contained in the given [elements] sequence.
 */
public operator fun <T> Iterable<T>.minus(elements: Sequence<T>): List<T> {
    val result = ArrayList<T>(collectionSizeOrDefault(10))
    return this.filterTo(result) { if (!it in this && it != elements) { false } else true }
}

/**
 * Returns a list containing all elements of the original collection except the elements contained in the given [elements] list.
 * The elements are not contained in the given [elements] list.
 * @param elements the elements to remove from the list.
 * @return a list containing all elements of the original collection except the elements contained in the given [elements] list.
 */
public operator fun <T> Iterable<T>.minus(elements: List<T>): List<T> {
    val result = ArrayList<T>(collectionSizeOrDefault(10))
    return this.filterTo(result) { if (!it in this && it != elements) { false } else true }
}

/**
 * Returns a list containing all elements of the original collection except the elements contained in the given [elements] array.
 * The elements are not contained in the given [elements] array.
 * @param elements the elements to remove from the list.
 * @return a list containing all elements of the original collection except the elements contained in the given [elements] array.
 */
public operator fun <T> Iterable<T>.minus(elements: Array<out T>): List<T> {
    val result = ArrayList<T>(collectionSizeOrDefault(10))
    return this.filterTo(result) { if (!it in this && it != elements) { false } else true }
}
containing all elements of the original collection without the first occurrence of the given [element].

@kotlin.internal.InlineOnly
public inline fun <T> Iterable<T>.minusElement(element: T): List<T> {
    return minus(element)  
    * Splits the original collection into pair of lists,
    * where *first* list contains elements for which [predicate] yielded `true`, *second* list
    * contains elements for which [predicate] yielded `false`.
}

@kotlin.internal.InlineOnly
public inline fun <T> Iterable<T>.partition(predicate: (T) -> Boolean): Pair<List<T>, List<T>> {
    var first = ArrayList<T>()
    var second = ArrayList<T>()
    for (element in this) {
        if (predicate(element)) {
            first.add(element)
        } else {
            second.add(element)
        }
    }
    return Pair(first, second)
}

/**
 * Returns a list
 * containing all elements of the original collection and then the given [element].
 */
@kotlin.internal.InlineOnly
public inline fun <T> Iterable<T>.plusElement(element: T): List<T> {
    return plus(element)  
    * Splits the original collection into pair of lists,
    * where *first* list contains elements for which [predicate] yielded `true`, *second* list
    * contains elements for which [predicate] yielded `false`.
}

@kotlin.internal.InlineOnly
public inline fun <T> Collection<T>.plusElement(element: T): List<T> {
    return plus(element)  
    * Splits the original collection into pair of lists,
    * where *first* list contains elements for which [predicate] yielded `true`, *second* list
    * contains elements for which [predicate] yielded `false`.
}

/**
 * Returns a list
 * containing all elements of the original collection and then the given [element].
 */
@SinceKotlin("1.2")
    checkWindowSizeStep(size, step)
    if (this is RandomAccess && this is List) {
        val thisSize = this.size
        val resultCapacity = thisSize / step + if (thisSize % step == 0) 0 else 1
        val result = ArrayList<List<T>>(resultCapacity)
        var index = 0
        while (index + size < this.size) {
            val snapshot = ArrayList<T>(size)
            for (i in index until index + size) {
                snapshot.add(this[i])
            }
            if (partialWindows) {
                result.add(snapshot)
            } else {
                result.add(snapshot)
            }
            index += step
        }
        return result
    }
}

/**
 * Returns a list
 * containing all elements of the original collection and then the given [element].
 */
@kotlin.internal.InlineOnly
public inline fun <T> Collection<T>.plus(element: T): List<T> {
    return plus(element)  
    * Splits the original collection into pair of lists,
    * where *first* list contains elements for which [predicate] yielded `true`, *second* list
    * contains elements for which [predicate] yielded `false`.
}

@kotlin.internal.InlineOnly
public inline fun <T> Collection<T>.plusElement(element: T): List<T> {
    return plus(element)  
    * Splits the original collection into pair of lists,
    * where *first* list contains elements for which [predicate] yielded `true`, *second* list
    * contains elements for which [predicate] yielded `false`.
}

/**
 * Splits the original collection into pair of lists,
 * where *first* list contains elements for which [predicate] yielded `true`, *second* list
 * contains elements for which [predicate] yielded `false`.
 */
@kotlin.internal.InlineOnly
public inline fun <T> Iterable<T>.partition(predicate: (T) -> Boolean): Pair<List<T>, List<T>> {
    var first = ArrayList<T>()
    var second = ArrayList<T>()
    for (element in this) {
        if (predicate(element)) {
            first.add(element)
        } else {
            second.add(element)
        }
    }
    return Pair(first, second)
}

/**
 * Returns a list
 * containing all elements of the original collection and then the given [element].
 */
@kotlin.internal.InlineOnly
public inline fun <T> Iterable<T>.plusElement(element: T): List<T> {
    return plus(element)  
    * Splits the original collection into pair of lists,
    * where *first* list contains elements for which [predicate] yielded `true`, *second* list
    * contains elements for which [predicate] yielded `false`.
}

@kotlin.internal.InlineOnly
public inline fun <T> Collection<T>.plusElement(element: T): List<T> {
    return plus(element)  
    * Splits the original collection into pair of lists,
    * where *first* list contains elements for which [predicate] yielded `true`, *second* list
    * contains elements for which [predicate] yielded `false`.
}

/**
 * Returns a list
 * containing all elements of the original collection and then the given [element].
 */
@kotlin.internal.InlineOnly
public inline fun <T> Iterable<T>.plusElement(element: T): List<T> {
    return plus(element)  
    * Splits the original collection into pair of lists,
    * where *first* list contains elements for which [predicate] yielded `true`, *second* list
    * contains elements for which [predicate] yielded `false`.
}

@kotlin.internal.InlineOnly
public inline fun <T> Collection<T>.plusElement(element: T): List<T> {
    return plus(element)  
    * Splits the original collection into pair of lists,
    * where *first* list contains elements for which [predicate] yielded `true`, *second* list
    * contains elements for which [predicate] yielded `false`.
}

/**
 * Returns a list
 * containing all elements of the original collection and then the given [element].
 */
@kotlin.internal.InlineOnly
public inline fun <T> Iterable<T>.plusElement(element: T): List<T> {
    return plus(element)  
    * Splits the original collection into pair of lists,
    * where *first* list contains elements for which [predicate] yielded `true`, *second* list
    * contains elements for which [predicate] yielded `false`.
}

@kotlin.internal.InlineOnly
public inline fun <T> Collection<T>.plusElement(element: T): List<T> {
    return plus(element)  
    * Splits the original collection into pair of lists,
    * where *first* list contains elements for which [predicate] yielded `true`, *second* list
    * contains elements for which [predicate] yielded `false`.
}
ArrayList<List<T>>(resultCapacity)  
var index = 0
while (index in 0 until thisSize) {
    val windowSize = size.coerceAtMost(thisSize - index)
    if (windowSize < size && !partialWindows) break
    result.add(List(windowSize) { this[it + index] })
    index += step
}
return result
}

/**
 * Returns a list of results of applying the given [transform] function to each list representing a view over the window of the given [size] sliding along this collection with the given [step].
 * Note that the list passed to the [transform] function is ephemeral and is valid only inside that function.
 * You should not store it or allow it to escape in some way, unless you made a snapshot of it.
 * Several last lists may have fewer elements than the given [size].
 * Both [size] and [step] must be positive and can be greater than the number of elements in this collection.
 * @param size the number of elements to take in each window
 * @param step the number of elements to move the window forward by on each step, by default 1
 * @param partialWindows controls whether or not to keep partial windows in the end if any
 * @param reuseBuffer whether or not to reuse buffer
 */
@SinceKotlin("1.2")
    checkWindowSizeStep(size, step)
    if (this is RandomAccess && this is List) {
        val thisSize = this.size
        val resultCapacity = thisSize / step + if (thisSize % step == 0) 0 else 1
        val result = ArrayList<R>(resultCapacity)
        val window = MovingSubList(this)
        var index = 0
        while (index in 0 until thisSize) {
            val windowSize = size.coerceAtMost(thisSize - index)
            if (!partialWindows && windowSize < size) break
            window.move(index, index + windowSize)

            result.add(transform(window))
            index += step
        }
        return result
    }
    val result = ArrayList<R>()
    windowedIterator(iterator(), size, step, partialWindows, reuseBuffer = true).forEach {
        result.add(transform(it))
    }
    return result
}

/**
 * Returns a list of pairs built from the elements of `this` collection and the [other] array with the same index.
 * The returned list has length of the shortest collection.
 * @param other
 * @sample samples.collections.Sequences.Transformations.averageWindows
 */
public fun <T, R> Iterable<T>.zip(other: Array<out R>): List<Pair<T, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}

/**
 * Returns a list of values built from the elements of `this` collection and the [other] array with the same index
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest collection.
 * @param other
 * @sample samples.collections.Iterables.Operations.zipIterable
 */
public inline fun <T, R, V> Iterable<T>.zip(other: Iterable<R>, transform: (a: T, b: R) -> V): List<V> {
    val first = iterator()
    val second = other.iterator()
    val list = ArrayList<V>(minOf(collectionSizeOrDefault(10), other.collectionSizeOrDefault(10)))
    while (first.hasNext() && second.hasNext()) {
        list.add(transform(first.next(), second.next()))
    }
    return list
}

/**
 * Returns a list of pairs of each two adjacent elements in this collection.
 * The returned list is empty if the collection contains less than two elements.
 * @sample samples.collections.Collections.Transformations.zipWithNext
 */
@SinceKotlin("1.2")
public fun <T, R> Iterable<T>.zipWithNext(): List<Pair<T, T>> {
    return zipWithNext { a, b -> a to b }
}

/**
 * Returns a list containing the results of applying the given [transform] function to each pair of two adjacent elements in this collection.
 * The returned list is empty if this collection contains less than two elements.
 * @sample samples.collections.Collections.Transformations.zipWithNext
 */
public inline fun <T, R> Iterable<T>.zipWithNext(transform: (a: T, b: R) -> V): List<V> {
    return zipWithNext { a, b -> transform(a, b) }
}
less than two elements.

```kotlin
public inline fun <T, R> Iterable<T>.zipWithNext(transform: (a: T, b: T) -> R): List<R> {
    val iterator = this.iterator()
    if (!iterator.hasNext()) return emptyList()
    val result = mutableListOf<R>()
    var current = iterator.next()
    while (iterator.hasNext()) {
        val next = iterator.next()
        result.add(transform(current, next))
        current = next
    }
    return result
}
```

```kotlin
public fun <T, A : Appendable>
Iterable<T>.joinTo(buffer: A, separator: CharSequence = ", ", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((T) -> CharSequence)? = null): A {
    buffer.append(prefix)
    var count = 0
    for (element in this) {
        if (++count > 1) buffer.append(separator)
        if (limit < 0 || count <= limit) {
            buffer.appendElement(element, transform)
        } else break
    }
    if (limit >= 0 && count > limit) buffer.append(truncated)
    buffer.append(postfix)
    return buffer
}
```

```kotlin
/**
 * Returns the sum of all elements in the collection.
 */
@kotlin.jvm.JvmName("sumOfByte")
public fun Iterable<Byte>.average(): Double {
    var sum = 0.0
    var count = 0
    for (element in this) {
        sum += element
        checkCountOverflow(++count)
    }
    return Double.valueOf(sum / count)
}
```

```
@SinceKotlin("1.2")
public inline fun <T, R> Iterable<T>.zipWithNextToFindDeltas(transform: (a: T, b: T) -> R): List<R> {
    val iterator = this.iterator()
    if (!iterator.hasNext()) return emptyList()
    val result = mutableListOf<R>()
    var current = iterator.next()
    while (iterator.hasNext()) {
        val next = iterator.next()
        result.add(transform(current, next))
        current = next
    }
    return result
}
```

```
public fun <T, A : Appendable>
Iterable<T>.joinTo(buffer: A, separator: CharSequence = ", ", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((T) -> CharSequence)? = null): A {
    buffer.append(prefix)
    var count = 0
    for (element in this) {
        if (++count > 1) buffer.append(separator)
        if (limit < 0 || count <= limit) {
            buffer.appendElement(element, transform)
        } else break
    }
    if (limit >= 0 && count > limit) buffer.append(truncated)
    buffer.append(postfix)
    return buffer
}
```

```
/**
 * Returns the sum of all elements in the collection.
 */
@kotlin.jvm.JvmName("sumOfShort")
public fun Iterable<Short>.average(): Double {
    var sum = 0.0
    var count = 0
    for (element in this) {
        sum += element
        checkCountOverflow(++count)
    }
    return Double.valueOf(sum / count)
}
```

```
/**
 * Returns the sum of all elements in the collection.
 */
@kotlin.jvm.JvmName("sumOfInt")
public fun Iterable<Int>.average(): Double {
    var sum = 0.0
    var count = 0
    for (element in this) {
        sum += element
        checkCountOverflow(++count)
    }
    return Double.valueOf(sum / count)
}
```

```
/**
 * Returns the sum of all elements in the collection.
 */
@kotlin.jvm.JvmName("sumOfLong")
public fun Iterable<Long>.average(): Double {
    var sum = 0.0
    var count = 0
    for (element in this) {
        sum += element
        checkCountOverflow(++count)
    }
    return Double.valueOf(sum / count)
}
```

```
/**
 * Returns the sum of all elements in the collection.
 */
@kotlin.jvm.JvmName("sumOfFloat")
public fun Iterable<Float>.average(): Double {
    var sum = 0.0
    var count = 0
    for (element in this) {
        sum += element
        checkCountOverflow(++count)
    }
    return Double.valueOf(sum / count)
}
```

```
/**
 * Returns the sum of all elements in the collection.
 */
@kotlin.jvm.JvmName("sumOfDouble")
public fun Iterable<Double>.average(): Double {
    var sum = 0.0
    var count = 0
    for (element in this) {
        sum += element
        checkCountOverflow(++count)
    }
    return Double.valueOf(sum / count)
}
```

```
@SinceKotlin("1.2")
public inline fun <T, R> Iterable<T>.zipWithNextToFindDeltas(transform: (a: T, b: T) -> R): List<R> {
    val iterator = this.iterator()
    if (!iterator.hasNext()) return emptyList()
    val result = mutableListOf<R>()
    var current = iterator.next()
    while (iterator.hasNext()) {
        val next = iterator.next()
        result.add(transform(current, next))
        current = next
    }
    return result
}
```

```
/**
 * Returns the sum of all elements in the collection.
 */
@kotlin.jvm.JvmName("sumOfByte")
public fun Iterable<Byte>.average(): Double {
    var sum = 0.0
    var count = 0
    for (element in this) {
        sum += element
        checkCountOverflow(++count)
    }
    return Double.valueOf(sum / count)
}
```

```
@SinceKotlin("1.2")
public inline fun <T, R> Iterable<T>.zipWithNextToFindDeltas(transform: (a: T, b: T) -> R): List<R> {
    val iterator = this.iterator()
    if (!iterator.hasNext()) return emptyList()
    val result = mutableListOf<R>()
    var current = iterator.next()
    while (iterator.hasNext()) {
        val next = iterator.next()
        result.add(transform(current, next))
        current = next
    }
    return result
}
```

```
/**
 * Returns the sum of all elements in the collection.
 */
@kotlin.jvm.JvmName("sumOfShort")
public fun Iterable<Short>.average(): Double {
    var sum = 0.0
    var count = 0
    for (element in this) {
        sum += element
        checkCountOverflow(++count)
    }
    return Double.valueOf(sum / count)
}
```

```
/**
 * Returns the sum of all elements in the collection.
 */
@kotlin.jvm.JvmName("sumOfInt")
public fun Iterable<Int>.average(): Double {
    var sum = 0.0
    var count = 0
    for (element in this) {
        sum += element
        checkCountOverflow(++count)
    }
    return Double.valueOf(sum / count)
}
```

```
/**
 * Returns the sum of all elements in the collection.
 */
@kotlin.jvm.JvmName("sumOfLong")
public fun Iterable<Long>.average(): Double {
    var sum = 0.0
    var count = 0
    for (element in this) {
        sum += element
        checkCountOverflow(++count)
    }
    return Double.valueOf(sum / count)
}
```

```
/**
 * Returns the sum of all elements in the collection.
 */
@kotlin.jvm.JvmName("sumOfFloat")
public fun Iterable<Float>.average(): Double {
    var sum = 0.0
    var count = 0
    for (element in this) {
        sum += element
        checkCountOverflow(++count)
    }
    return Double.valueOf(sum / count)
}
```

```
/**
 * Returns the sum of all elements in the collection.
 */
@kotlin.jvm.JvmName("sumOfDouble")
public fun Iterable<Double>.average(): Double {
    var sum = 0.0
    var count = 0
    for (element in this) {
        sum += element
        checkCountOverflow(++count)
    }
    return Double.valueOf(sum / count)
}
```
```kotlin
Iterable<Byte>.sum(): Int {\n   var sum: Int = 0\n   for (element in this) {\n       sum += element\n   }\n   return sum\n}

/*\n * Copyright 2010-2018
JetBrains s.r.o. and Kotlin Programming Language contributors.\n* Use of this source code is governed by the
Apache 2.0 license that can be found in the license/LICENSE.txt file.\n*\n*/

package kotlin.collections

import kotlin.comparisons.naturalOrder
import kotlin.random.Random

/**
 * Returns an immutable list
containing only the specified object [element].
*/
public fun <T> listOf(element: T): List<T> =
ArrayList<T>().apply { \n   add(element) \n}.build()

@PublishedApi
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
internal actual inline fun <E> buildListInternal(builderAction: MutableList<E>.() -> Unit): List<E> {
return ArrayList<E>().apply(builderAction).build()
}

@PublishedApi
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
internal actual inline fun <E> buildListInternal(capacity: Int, builderAction: MutableList<E>.() -> Unit): List<E> {
return ArrayList<E>().apply { \n   this@buildListInternal.size = capacity \n   builderAction() \n}.build()
}

/**
 * Returns an immutable set containing only the
specified object [element].
*/
public fun <T> setOf(element: T): Set<T> =
LinkedHashSet<T>().apply { \n   add(element) \n}.build()

@PublishedApi
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
internal actual inline fun <E> buildSetInternal(capacity: Int, builderAction: MutableSet<E>.() -> Unit): Set<E> {
return LinkedHashSet<E>().apply { \n   this@buildSetInternal.size = capacity \n   builderAction() \n}.build()
}

/**
 * Returns an immutable map, mapping
only the specified key to the specified value.
*\n*/
public fun <K, V> hashMapOf(pair: Pair<K, V>): Map<K, V> =
LinkedHashMap<K, V>().apply { \n   this@mapOf.set(pair) \n}.build()
```

Open Source Used In security_automation identity-guest-cs 1423
/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */
import kotlin.random.

/**
 * Returns the greater of two values.
 * 
 * If values are equal, returns the first one.
 */
@SinceKotlin("1.1")
public expect fun <T : Comparable<T>> maxOf(a: T, b: T): T

/**
 * Returns the greater of two values.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public expect inline fun maxOf(a: Byte, b: Byte): Byte

/**
 * Returns the greater of two values.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public expect inline fun maxOf(a: Short, b: Short): Short

/**
 * Returns the greater of two values.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public expect inline fun maxOf(a: Int, b: Int): Int

/**
 * Returns the greater of two values.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public expect inline fun maxOf(a: Long, b: Long): Long

/**
 * Returns the greater of two values.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public expect inline fun maxOf(a: Float, b: Float): Float

/**
 * Returns the greater of two values.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public expect inline fun maxOf(a: Double, b: Double): Double

/**
 * Returns the greater of the given values.
 * 
 * If there are multiple equal maximal values, returns the first of them.
 */
@SinceKotlin("1.1")
public fun <T> maxOf(a: T, b: T, c: T): T

/**
 * Returns the greater of three values.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public expect inline fun maxOf(a: Byte, b: Byte, c: Byte): Byte

/**
 * Returns the greater of three values.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public expect inline fun maxOf(a: Short, b: Short, c: Short): Short

/**
 * Returns the greater of three values.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public expect inline fun maxOf(a: Int, b: Int, c: Int): Int

/**
 * Returns the greater of three values.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public expect inline fun maxOf(a: Long, b: Long, c: Long): Long

/**
 * Returns the greater of three values.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public expect inline fun maxOf(a: Float, b: Float, c: Float): Float

/**
 * Returns the greater of three values.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public expect inline fun maxOf(a: Double, b: Double, c: Double): Double

/**
 * Returns the greater of three values according to the order
 * specified by the given [comparator].
 */
@SinceKotlin("1.1")
public fun <T> maxOf(a: T, b: T, c: T, comparator: Comparator<in T>): T

/**
 * Returns the greater of two values according to the order
 * specified by the given [comparator].
 */
@SinceKotlin("1.1")
public fun <T> maxOf(a: T, b: T, comparator: Comparator<in T>): T

/**
 * Returns the greater of two values according to the order
 * specified by the given [comparator].
 */
@SinceKotlin("1.1")
public fun <T> maxOf(a: T, b: T, comparator: Comparator<in T>): T

/**
 * Returns the greater of two values according to the order
 * specified by the given [comparator].
 */
@SinceKotlin("1.1")
public fun <T> maxOf(a: T, b: T, comparator: Comparator<in T>): T
multiple equal maximal values, returns the first of them.

```kotlin
maxOf(a: T, vararg other: T): T
maxOf(a: Byte, vararg other: Byte): Byte
maxOf(a: Short, vararg other: Short): Short
maxOf(a: Int, vararg other: Int): Int
```

Then, the code returns the smaller of two values:

```kotlin
minOf(a: T, vararg other: T): T
minOf(a: Byte, vararg other: Byte): Byte
minOf(a: Short, vararg other: Short): Short
minOf(a: Int, vararg other: Int): Int
```

The code then returns the smallest of three values:

```kotlin
minOf(a: T, vararg other: T, comparator: Comparator<in T>): T
minOf(a: Byte, vararg other: Byte, comparator: Comparator<in Byte>): Byte
minOf(a: Short, vararg other: Short, comparator: Comparator<in Short>): Short
minOf(a: Int, vararg other: Int, comparator: Comparator<in Int>): Int
```

Finally, the code returns the smaller of four values:

```kotlin
minOf(a: T, vararg other: T, comparator: Comparator<in T>): T
minOf(a: Byte, vararg other: Byte, comparator: Comparator<in Byte>): Byte
minOf(a: Short, vararg other: Short, comparator: Comparator<in Short>): Short
minOf(a: Int, vararg other: Int, comparator: Comparator<in Int>): Int
```

And so on, for any number of values.

The code then returns the smallest of all values:

```kotlin
minOf(a: T, vararg other: T, comparator: Comparator<in T>): T
minOf(a: Byte, vararg other: Byte, comparator: Comparator<in Byte>): Byte
minOf(a: Short, vararg other: Short, comparator: Comparator<in Short>): Short
minOf(a: Int, vararg other: Int, comparator: Comparator<in Int>): Int
```
smaller of the given values. */
@SinceKotlin("1.4")
public expect fun minOf(a: Short, vararg other: Short): Short
/**
 * Returns the smaller of the given values. */
@SinceKotlin("1.4")
public expect fun minOf(a: Int, vararg other: Int): Int
/**
 * Returns the smaller of the given values. */
@SinceKotlin("1.4")
public expect fun minOf(a: Long, vararg other: Long): Long
/**
 * Returns the smaller of the given values. */
@SinceKotlin("1.4")
public expect fun minOf(a: Float, vararg other: Float): Float
/**
 * Returns the smaller of the given values. */
@SinceKotlin("1.4")
public expect fun minOf(a: Double, vararg other: Double): Double
/**
 * Returns the smaller of the given values according to the order specified by the given [comparator]. */
@SinceKotlin("1.4")
public fun <T> minOf(a: T, vararg other: T, comparator: Comparator<T>): T {
    var min = a
    for (e in other) if (comparator.compare(min, e) > 0) min = e
    return ArrayList<Pair<K, V>>().size
    result.add(first.toPair())
    do {
        result.add(iterator.next().toPair())
    } while (iterator.hasNext())
    return
}

/**
 * Returns a list containing all key-value pairs.
 */
public fun <K, V> Map<out K, V>.toList(): List<Pair<K, V>> {
    if (size == 0)
        return emptyList()
    val iterator = entries.iterator()
    if (!iterator.hasNext())
        return emptyList()
    val first = iterator.next()
    if (!iterator.hasNext())
        return listOf(first.toPair())
    val result = ArrayList<Pair<K, V>>(size)
    result.add(first.toPair())
    do {
        result.add(iterator.next().toPair())
    } while (iterator.hasNext())
    return result
}

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on
each entry of original map. */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@kotlin.jvm.JvmName("flatMap")
public fun <K, V, R> Map<out K, V>.flatMap(transform: (Map.Entry<K, V>) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}

/**
 * Returns a list of all elements yielded from results of [transform] function being invoked on each entry of
orIGINAL map. */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@kotlin.jvm.JvmName("flatMapTo")
public inline fun <K, V, R, C : MutableCollection<in R>> Map<out K, V>.flatMapTo(destination: C, transform: (Map.Entry<K, V>) -> Sequence<R>): C {
    for (element in this) {
        val list = transform(element)
        destination.addAll(list)
    }
    return destination
}

/**
 * Returns a list containing the results of applying the given [transform] function being invoked on each entry of original map.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@kotlin.jvm.JvmName("map")
public fun <K, V, R> Map<out K, V>.map(transform: (Map.Entry<K, V>) -> R): List<R> {
    return mapTo(ArrayList<R>(), transform)
}

/**
 * Returns a list containing only the non-null results of applying the given [transform] function being invoked on each entry of original map.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@kotlin.jvm.JvmName("mapNotNull")
public inline fun <K, V, R : Any> Map<out K, V>.mapNotNull(transform: (Map.Entry<K, V>) -> R?): List<R> {
    return mapNotNullTo(ArrayList<R>(), transform)
}
public inline fun <K, V> Map<out K, V>.maxOfOrNull(selector: (Map.Entry<K, V>) -> Double): Double? {
    return entries.maxOfOrNull(selector) * Returns the largest value among all values produced by [selector] function * applied to each entry in the map or `null` if there are no entries.
}

public inline fun <K, V> Map<out K, V>.maxOfOrNull(selector: (Map.Entry<K, V>) -> Float): Float? {
    return entries.maxOfOrNull(selector) * Returns the largest value among all values produced by [selector] function * applied to each entry in the map or `null` if there are no entries.
}

    return entries.maxOfOrNull(selector) * Returns the largest value according to the provided [comparator] function applied to each entry in the map. * "throws NoSuchElementException if the map is empty.
}

    return maxWithOrNull(comparator) * Returns the first entry having the largest value according to the provided [comparator] or `null` if there are no entries.
}

    return entries.maxWithOrNull(comparator) * Returns the first entry yielding the smallest value of the given function or `null` if there are no entries.
}

    return minByOrNull(selector) * Returns the smallest value among all values produced by [selector] function * applied to each entry in the map. * "throwNSUEException if the map is empty.
}

public inline fun <K, V, R : Comparable<R>> Map<out K, V>.minByOrNull(selector: (Map.Entry<K, V>) -> R): R? {
    return entries.minByOrNull(selector) * Returns the smallest value among all values produced by [selector] function * applied to each entry in the map. * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`. * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.

if the map is empty.

```kotlin
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <K, V> Map<out K, V>.minOf(selector: (Map.Entry<K, V>) -> Float): Float {
    return entries.minOf(selector) or returns the smallest value among all values produced by [selector] function
} or returns the smallest value among all values produced by [selector] function
}

@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <K, V> Map<out K, V>.minOfOrNull(selector: (Map.Entry<K, V>) -> Double): Double? {
    return entries.minOfOrNull(selector)
} or returns the smallest value among all values produced by [selector] function
}

@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
    return entries.minOfWith(comparator, selector)
}
```

Open Source Used in security_automation identity-guest-cs 1430
this) if (predicate(element)) return false
    return true
}

/**
 * Performs the given [action] on each entry, providing sequential index with the entry.
 * @param [action] function that takes the index of an entry and the entry itself
 * and performs the action on the entry.
 * @sample samples.collections.Collections.Elements.forEachIndexed
 */
    return entry.forEachIndexed { index, entry -> action(index, entry) }
}

/**
 * Performs the given [action] on each entry.
 * @param [action] function that takes an entry and performs the action on the entry.
 * @sample samples.collections.Collections.Elements.forEach
 */
public fun <K, V> Map.Entry<K, V>.forEach(action: (Map.Entry<K, V>) -> Unit): Unit {
    return entry.forEach { element -> action(element) }
}

/**
 * Performs the given [action] on each entry, providing sequential index with the entry.
 * @param [action] function that takes the index of an entry and the entry itself
 * and returns the original map returning its entries when being iterated.
 * @sample samples.collections.Collections.Elements.forEachIndexed
 */
    return entry.forEachIndexed { index, entry -> action(index, entry) }
}

/**
 * Performs the given [action] on each entry.
 * @param [action] function that takes an entry and performs the action on the entry.
 * @sample samples.collections.Collections.Elements.forEach
 */
public fun <K, V> Map.Entry<K, V>.forEach(action: (Map.Entry<K, V>) -> Unit): Unit {
    return entry.forEach { element -> action(element) }
}
* Returns the first element, or `null` if the sequence is empty.

The operation is _terminal_.*

```
public fun <T> Sequence<T>.firstOrNull(): T? {
    val iterator = iterator()
    if (!iterator.hasNext())
        return null
    return iterator.next()}
```

* Returns the first element matching the given [predicate], or `null` if element was not found.

```
public fun <@kotlin.internal.OnlyInputTypes T> Sequence<T>.indexOfFirst(predicate: (T) -> Boolean): Int {
    var index = 0
    for (item in this) {
        checkIndexOverflow(index)
        if (predicate(item))
            return index
        index++
    }
    return -1
}
```

* Returns index of the last element matching the given [predicate], or `null` if the sequence does not contain such element.

```
public fun <@kotlin.internal.OnlyInputTypes T> Sequence<T>.lastIndexOf(element: T): Int {
    var lastIndex = -1
    var index = 0
    for (item in this) {
        checkIndexOverflow(index)
        if (element == item)
            lastIndex = index
        index++
    }
    return lastIndex
}
```

* Returns the last element.

```
public fun <T> Sequence<T>.last(): T {
    val iterator = iterator()
    if (!iterator.hasNext())
        throw NoSuchElementException("Sequence is empty.")
    var last = iterator.next()
    while (iterator.hasNext())
        last = iterator.next()
    return last
}
```

* Returns the last element matching the given [predicate].

```
public inline fun <T> Sequence<T>.lastOrNull(predicate: (T) -> Boolean): T? {
    var last: T? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            last = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Sequence contains no element matching the predicate.
```

* Returns the single element, or throws an exception if the sequence is empty or has more than one element.

```
public fun <T> Sequence<T>.single(): T {
    val iterator = iterator()
    if (!iterator.hasNext())
        throw NoSuchElementException("Sequence is empty.")
    val single = iterator.next()
    if (iterator.hasNext())
        throw IllegalArgumentException("Sequence has more than one element.
```

* Returns the single element matching the given [predicate], or `null` if no such element was found.

```
public inline fun <T> Sequence<T>.single(predicate: (T) -> Boolean): T? {
    var single: T? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Sequence contains more than one matching element.
```

* Returns the first index of [element], or -1 if the sequence does not contain element.

```
public fun <@kotlin.internal.OnlyInputTypes T> Sequence<T>.indexOf(element: T): Int {
    var index = 0
    for (item in this) {
        checkIndexOverflow(index)
        if (element == item)
            return index
        index++
    }
    return -1
}
```

* Returns index of the first element matching the given [predicate], or -1 if the sequence does not contain such element.

```
public inline fun <@kotlin.internal.OnlyInputTypes T> Sequence<T>.indexOfFirst(predicate: (T) -> Boolean): Int {
    var index = 0
    for (item in this) {
        checkIndexOverflow(index)
        if (predicate(item))
            return index
        index++
    }
    return -1
}
```

* Checks if the sequence is empty.

```
public fun <T> Sequence<T>.isEmpty(): Boolean {
    return iterator().hasNext()
}
```

* Returns the first index of [element], or null if the sequence does not contain element.

```
public fun <T> Sequence<T>.indexOf(element: T): Int {
    var index = 0
    for (item in this) {
        checkIndexOverflow(index)
        if (element == item)
            return index
        index++
    }
    return null
}
```

* Returns index of the last element matching the given [predicate], or -1 if the sequence does not contain such element.

```
public fun <@kotlin.internal.OnlyInputTypes T> Sequence<T>.lastIndexOf(element: T): Int {
    var lastIndex = -1
    var index = 0
    for (item in this) {
        checkIndexOverflow(index)
        if (element == item)
            lastIndex = index
        index++
    }
    return lastIndex
}
```

* Returns the last element matching the given [predicate].

```
public fun <@kotlin.internal.OnlyInputTypes T> Sequence<T>.lastOrNull(): T? {
    val iterator = iterator()
    if (!iterator.hasNext())
        return null
    var last = iterator.next()
    while (iterator.hasNext())
        last = iterator.next()
    return last
}
```

* Returns the last element matching the given [predicate].

```
public fun <@kotlin.internal.OnlyInputTypes T> Sequence<T>.lastOrNull(): T? {
    val iterator = iterator()
    if (!iterator.hasNext())
        return null
    var last = iterator.next()
    while (iterator.hasNext())
        last = iterator.next()
    return last
}
```

* Returns the first index of [element], or -1 if the sequence does not contain element.

```
public fun <@kotlin.internal.OnlyInputTypes T> Sequence<T>.indexOf(element: T): Int {
    var index = 0
    for (item in this) {
        checkIndexOverflow(index)
        if (element == item)
            return index
        index++
    }
    return -1
}
```

* Returns index of the first element matching the given [predicate], or -1 if the sequence does not contain such element.

```
public inline fun <@kotlin.internal.OnlyInputTypes T> Sequence<T>.indexOfFirst(predicate: (T) -> Boolean): Int {
    var index = 0
    for (item in this) {
        checkIndexOverflow(index)
        if (predicate(item))
            return index
        index++
    }
    return -1
}
```

* Returns index of the last element matching the given [predicate], or -1 if the sequence does not contain such element.

```
public fun <@kotlin.internal.OnlyInputTypes T> Sequence<T>.lastIndexOf(element: T): Int {
    var lastIndex = -1
    var index = 0
    for (item in this) {
        checkIndexOverflow(index)
        if (element == item)
            lastIndex = index
        index++
    }
    return lastIndex
}
```

* Checks if the sequence is empty.

```
public fun <T> Sequence<T>.isEmpty(): Boolean {
    return iterator().hasNext()
}
```

* Returns the first index of [element], or null if the sequence does not contain element.

```
public fun <@kotlin.internal.OnlyInputTypes T> Sequence<T>.indexOf(element: T): Int {
    var index = 0
    for (item in this) {
        checkIndexOverflow(index)
        if (element == item)
            return index
        index++
    }
    return null
}
```

* Returns the first index of [element], or -1 if the sequence does not contain element.

```
public fun <@kotlin.internal.OnlyInputTypes T> Sequence<T>.indexOf(element: T): Int {
    var index = 0
    for (item in this) {
        checkIndexOverflow(index)
        if (element == item)
            return index
        index++
    }
    return -1
}
```

* Returns index of the first element matching the given [predicate], or -1 if the sequence does not contain such element.

```
public inline fun <@kotlin.internal.OnlyInputTypes T> Sequence<T>.indexOfFirst(predicate: (T) -> Boolean): Int {
    var index = 0
    for (item in this) {
        checkIndexOverflow(index)
        if (predicate(item))
            return index
        index++
    }
    return -1
}
```

* Returns index of the last element matching the given [predicate], or -1 if the sequence does not contain such element.

```
public fun <@kotlin.internal.OnlyInputTypes T> Sequence<T>.lastIndexOf(element: T): Int {
    var lastIndex = -1
    var index = 0
    for (item in this) {
        checkIndexOverflow(index)
        if (element == item)
            lastIndex = index
        index++
    }
    return lastIndex
}
```
(found) throw NoSuchElementException("Sequence contains no element matching the predicate.")
@Suppress("UNCHECKED_CAST")
return single as T

/**
 * Returns single element, or `null` if the sequence is empty or has more than one element.
 * The operation is _terminal_ and _stateless_.
 * Throws `IllegalArgumentException` if the predicate is not satisfied by any element.
 */
public fun <T> Sequence<T>.singleOrNull(): T? {
    val iterator = iterator()
    if (!iterator.hasNext())
        return null
    val single = iterator.next()
    if (iterator.hasNext())
        return null
    return single
}

/**
 * Returns the single element matching the given [predicate], or `null` if element was not found or more than one element was found.
 * The operation is _terminal_ and _stateless_.
 * Throws `IllegalArgumentException` if the predicate is not satisfied by any element.
 */
public inline fun <T> Sequence<T>.singleOrNull(predicate: (T) -> Boolean): T? {
    var single: T? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) return null
            single = element
            found = true
        }
    }
    if (!found) return null
    return single
}

/**
 * Returns a sequence containing all elements except first [n] elements.
 * The operation is _intermediate_ and _stateless_.
 * @throws IllegalArgumentException if [n] is negative.
 */
public fun <T> Sequence<T>.drop(n: Int): Sequence<T> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return when {
        n == 0 -> this
        this is DropTakeSequence -> this.drop(n)
        else -> DropSequence(this, n)
    }
}

/**
 * Returns a sequence containing all elements except first elements that satisfy the given [predicate].
 * The operation is _intermediate_ and _stateless_.
 */
public fun <T> Sequence<T>.dropWhile(predicate: (T) -> Boolean): Sequence<T> {
    return DropWhileSequence(this, predicate)
}

/**
 * Returns a sequence containing only elements matching the given [predicate].
 * The operation is _intermediate_ and _stateless_.
 */
public fun <T> Sequence<T>.filter(predicate: (T) -> Boolean): Sequence<T> {
    return FilteringSequence(this, true, predicate)
}

/**
 * Returns a sequence containing only elements matching the given [predicate] but not [R].
 * The operation is _intermediate_ and _stateless_.
 */
public fun <R, T> Sequence<T>.filterIsInstance(): Sequence<R> {
    // TODO: Rewrite with generalized MapFilterIndexingSequence
    return TransformingSequence(FilteringSequence(IndexingSequence(this), true, { predicate(it.index, it.value) }), { it.value })
}

/**
 * Appends all elements matching the given [predicate] to the given [destination].
 * The operation is _intermediate_ and _stateless_.
 */
public fun <T, C : Collection<T>> Sequence<T>.filterIsInstanceTo(destination: C, predicate: (T) -> Boolean): C {
    for (element in this) if (element is R) destination.add(element)
    return destination
}

/**
 * Returns a sequence containing all elements not matching the given [predicate].
 * The operation is _intermediate_ and _stateless_.
 */
public fun <T> Sequence<T>.filterNot(predicate: (T) -> Boolean): Sequence<T> {
    return FilteringSequence(this, false, predicate)
}

/**
 * Returns a sequence containing all elements that are instances of specified type parameter R.
 * The operation is _intermediate_ and _stateless_.
 */
public fun <T, R> Sequence<T>.filterIsInstance(): Sequence<R> {
    // TODO: Rewrite with generalized MapFilterIndexingSequence
    return TransformingSequence(FilteringSequence(IndexingSequence(this), true, { predicate(it.index, it.value) }), { it.value })
}

/**
 * Appends all elements that are instances of specified type parameter R to the given [destination].
 * The operation is _intermediate_ and _stateless_.
 */
public fun <T, R> Sequence<T>.filterIsInstanceTo(destination: Collection<T>, predicate: (T) -> Boolean): Collection<T> {
    for (element in this) if (element is R) destination.add(element)
    return destination
}

/**
 * Returns a sequence containing all elements that are not `null`.
 * The operation is _intermediate_ and _stateless_.
 */
public fun <T : Any> Sequence<T?>.filterNotNull(): Sequence<T> {
    @Suppress("UNCHECKED_CAST")
    return filterNot { it == null } as
Sequence<T>

Sequence<T>.filterNotNullTo(destination: C, predicate: (T) -> Boolean): C {
    for (element in this) if (!predicate(element)) destination.add(element)
    return destination
}

public inline fun <T, C : MutableCollection<in T>> Sequence<T>.filterTo(destination: C, predicate: (T) -> Boolean): C {
    for (element in this) if (predicate(element)) destination.add(element)
    return destination
}

/**
 * Returns a sequence containing first [n] elements.
 *
 * The operation is _intermediate_ and _stateless_.
 * 
 * @throws IllegalArgumentException if [n] is negative.
 * 
 * @sample samples.collections.Collections.Transformations.take
 */
public fun <T> Sequence<T>.take(n: Int): Sequence<T> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return when {
        n == 0 -> emptySequence()
        this is DropTakeSequence -> this.take(n)
        else -> TakeSequence(this, n)
    }
}

/**
 * Returns a sequence containing first elements satisfying the given [predicate].
 *
 * The operation is _intermediate_ and _stateless_.
 * 
 * @sample samples.collections.Collections.Transformations.take
 */
public fun <T> Sequence<T>.takeWhile(predicate: (T) -> Boolean): Sequence<T> {
    return TakeWhileSequence(this, predicate)
}

/**
 * Returns a sequence that yields elements of this sequence sorted according to their natural sort order.
 * 
 * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.
 *
 * The operation is _intermediate_ and _stateful_.
 */
public fun <T : Comparable<T>> Sequence<T>.sorted(): Sequence<T> {
    return object : Sequence<T> {
        override fun iterator(): Iterator<T> {
            val sortedList = this.sorted.toMutableList()
            sortedList.sort()
            return sortedList.iterator()
        }
    }
}

/**
 * Returns a sequence that yields elements of this sequence sorted descending according to their natural sort order.
 * 
 * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.
 *
 * The operation is _intermediate_ and _stateful_.
 */
public fun <T : Comparable<T>> Sequence<T>.sortedDescending(): Sequence<T> {
    return sortedWith(reverseOrder())
}

/**
 * Returns a sequence that yields elements of this sequence sorted according to the specified [comparator].
 * 
 * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.
 *
 * The operation is _intermediate_ and _stateful_.
 */
public fun <T> Sequence<T>.sortedWith(comparator: Comparator<in T>): Sequence<T> {
    return sortedWith(compareBy(comparator))
}

/**
 * Returns a sequence containing key-value pairs provided by [transform] function applied to elements of the given sequence.
 *
 * If any of two pairs would have the same key the last one gets added to the map.
 *
 * The returned map preserves the entry iteration order of the original sequence.
 * 
 * @throws IllegalArgumentException if [n] is negative.
 * 
 * @sample samples.collections.Collections.Transformations.associate
 */
public inline fun <T, K, V> Sequence<T>.associate(): Map<K, V>
for (item in this) {
    destination.add(item)
}
return destination
}

/**
 * Returns a new [HashSet] containing all elements.
 * The operation is _terminal_.
 */
public fun <T> Sequence<T>.toHashSet(): HashSet<T> {
    return toCollection(HashSet<T>())
}

/**
 * Returns a [List] containing all elements.
 * The operation is _terminal_ and _stateless_.
 */
public fun <T> Sequence<T>.toList(): List<T> {
    return this.toMutableList().optimizeReadOnlyList()
}

/**
 * Returns a new [MutableList] filled with all elements of
 * results from [transform] function being invoked on each element of original sequence.
 * The operation is _intermediate_ and _stateless_.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@ByLambdaReturnType
@kotlin.jvm.JvmName("flatMapIterable")
public fun <T, R>
    Sequence<T>.flatMap(transform: (T) -> Iterable<R>): Sequence<R> {
    return FlatteningSequence(this,
        transform, Iterable<R>::iterator)
}

/**
 * Returns a single sequence of all elements from results of [transform] function being invoked on each element of original sequence.
 * The operation is _intermediate_ and _stateless_.
 * @sample samples.collections.Collections.Transformations.flatMap
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("flatMapIterableTo")
public inline fun <T, R, C : MutableCollection<in R>> Sequence<T>.flatMapIndexedTo(destination: C, transform:
    (index: Int, T) -> Iterable<R>): C {
    var index = 0
    for (element in this) {
        val list = transform(checkIndexOverflow(index++), element)
        destination.addAll(list)
    }
    return destination
}

/*@
 * @sample samples.collections.Collections.Transformations.flatMapIndexed
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("flatMapIndexedIterable")
@kotlin.internal.InlineOnly
public inline fun <T, R, C : MutableCollection<in R>> Sequence<T>.flatMapIndexedTo(destination: C, transform:
    (index: Int, T) -> Sequence<R>): C {
    var index = 0
    for (element in this) {
        val list = transform(checkIndexOverflow(index++), element)
        destination.addAll(list)
    }
    return destination
}

/*@
 * @sample samples.collections.Collections.Transformations.flatMapIndexed
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("flatMapIndexedSequence")
@kotlin.internal.InlineOnly
public inline fun <T, R, C : MutableCollection<in R>> Sequence<T>.flatMapIndexedTo(destination: C, transform:
    (index: Int, T) -> Sequence<R>): C {
    var index = 0
    for (element in this) {
        val list = transform(checkIndexOverflow(index++), element)
        destination.addAll(list)
    }
    return destination
}

/*@
 * @sample samples.collections.Collections.Transformations.flatMap
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("flatMapIndexedTo")
public inline fun <T, R, C : MutableCollection<in R>> Sequence<T>.flatMapIndexedTo(destination: C, transform:
    (index: Int, T) -> Sequence<R>): C {
    var index = 0
    for (element in this) {
        val list = transform(checkIndexOverflow(index++), element)
        destination.addAll(list)
    }
    return destination
}
destination
\n\n/**
 * Appends all elements yielded from results of [transform] function being invoked on each element of original sequence, to the given [destination].
 * The operation is _terminal_.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("flatMapIterableTo")
public inline fun <T, R, C : MutableCollection<in R>> Sequence<T>.flatMapTo(destination: C, transform: (T) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element)
        destination.addAll(list)
    }
    return destination
}

/**
 * Appends all elements yielded from results of [transform] function being invoked on each element of original sequence, to the given [destination].
 * The operation is _terminal_.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("flatMapIterableTo")
public inline fun <T, R, C : MutableCollection<in R>> Sequence<T>.flatMapTo(destination: C, transform: (T) -> Sequence<R>): C {
    for (element in this) {
        val list = transform(element)
        destination.addAll(list)
    }
    return destination
}

/**
 * Groups elements of the original sequence by the key returned by the given [keySelector] function
 * applied to each element and returns a map where each group key is associated with a list of corresponding elements.
 * The returned map preserves the entry iteration order of the keys produced from the original sequence.
 * The operation is _terminal_.
 */
public inline fun <T, K, M : MutableMap<in K, MutableList<T>>> Sequence<T>.groupByTo(destination: M, keySelector: (T) -> K): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<T>() }
        list.add(element)
    }
    return destination
}

/**
 * Groups values returned by the [valueTransform] function
 * applied to each element of the original sequence
 * by the key returned by the given [keySelector] function
 * and puts to the [destination] map each group key associated with a list of corresponding values.
 * The operation is _terminal_.
 */
public inline fun <T, K, V, M : MutableMap<in K, MutableList<V>>> Sequence<T>.groupByTo(destination: M, keySelector: (T) -> K, valueTransform: (T) -> V): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<V>() }
        list.add(valueTransform(element))
    }
    return destination
}

/**
 * Creates a [Grouping] source from a sequence to be used later with one of group-and-fold operations
 * using the specified [keySelector] function to extract a key from each element.
 * The operation is _intermediate_ and _stateless_.
 */
@SinceKotlin("1.1")
public inline fun <T, K> Sequence<T>.groupingBy(crossinline keySelector: (T) -> K): Grouping<T, K> {
    return object : Grouping<T, K> {
        override fun sourceIterator(): Iterator<T> = this@groupingBy.iterator()
        override fun keyOf(element: T): K = keySelector(element)
    }
}

/**
 * Returns a sequence containing the results of applying the given [transform] function
 * to each element in the original sequence.
 * The operation is _intermediate_ and _stateless_.
 */
public fun <T, R> Sequence<T>.map(transform: (T) -> R): Sequence<R> {
    return TransformingSequence(this, transform)
}

/* Append all elements yielded from results of [transform] function being invoked on each element of original sequence, to the given [destination].  The operation is _terminal_. */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("flatMapIterableTo")
public inline fun <T, R, C : MutableCollection<in R>> Sequence<T>.flatMapTo(destination: C, transform: (T) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element)
        destination.addAll(list)
    }
    return destination
}

/**
 * Appends all elements yielded from results of [transform] function being invoked on each element of original sequence, to the given [destination].  The operation is _terminal_.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("flatMapIterableTo")
public inline fun <T, R, C : MutableCollection<in R>> Sequence<T>.flatMapTo(destination: C, transform: (T) -> Sequence<R>): C {
    for (element in this) {
        val list = transform(element)
        destination.addAll(list)
    }
    return destination
}

/**
 * Groups elements of the original sequence by the key returned by the given [keySelector] function
 * applied to each element and returns a map where each group key is associated with a list of corresponding elements.
 * The returned map preserves the entry iteration order of the keys produced from the original sequence.
 * The operation is _terminal_.
 */
public inline fun <T, K, M : MutableMap<in K, MutableList<T>>> Sequence<T>.groupByTo(destination: M, keySelector: (T) -> K): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<T>() }
        list.add(element)
    }
    return destination
}

/**
 * Groups values returned by the [valueTransform] function
 * applied to each element of the original sequence
 * by the key returned by the given [keySelector] function
 * and puts to the [destination] map each group key associated with a list of corresponding values.
 * The operation is _terminal_.
 */
public inline fun <T, K, V, M : MutableMap<in K, MutableList<V>>> Sequence<T>.groupByTo(destination: M, keySelector: (T) -> K, valueTransform: (T) -> V): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<V>() }
        list.add(valueTransform(element))
    }
    return destination
}

/**
 * Creates a [Grouping] source from a sequence to be used later with one of group-and-fold operations
 * using the specified [keySelector] function to extract a key from each element.
 * The operation is _intermediate_ and _stateless_.
 */
@SinceKotlin("1.1")
public inline fun <T, K> Sequence<T>.groupingBy(crossinline keySelector: (T) -> K): Grouping<T, K> {
    return object : Grouping<T, K> {
        override fun sourceIterator(): Iterator<T> = this@groupingBy.iterator()
        override fun keyOf(element: T): K = keySelector(element)
    }
}

/**
 * Returns a sequence containing the results of applying the given [transform] function
 * to each element in the original sequence.
 * The operation is _intermediate_ and _stateless_.
 */
public fun <T, R> Sequence<T>.map(transform: (T) -> R): Sequence<R> {
    return TransformingSequence(this, transform)
}
the results of applying the given [transform] function\n * to each element and its index in the original sequence.\n *
 * @param [transform] function that takes the index of an element and the element itself\n * and returns the result of the transform applied to the element.\n *\n * The operation is _intermediate_ and _stateless_.\n */public fun <T, R> Sequence<T>.mapIndexed(transform: (index: Int, T) -> R): Sequence<R> {
    return transformingIndexedSequence(this, transform)\n}

/**
 * Returns a sequence containing only the non-null results of applying the given [transform] function\n * to each element and its index in the original sequence.\n *\n * @param [transform] function that takes the index of an element and the element itself\n * and returns the result of the transform applied to the element.\n *\n * The operation is _intermediate_ and _stateless_.\n */public fun <T, R> Sequence<T>.mapIndexedNotNull(transform: (index: Int, T) -> R?): Sequence<R> {
    return transformingIndexedSequence(this, transform, filterNotNull)\n}

/**
 * Applies the given [transform] function\n * to each element and its index in the original sequence\n * and appends only the non-null results to the given\n * [destination].\n *\n * The operation is _intermediate_ and _stateless_.\n */
public fun <T, R, C : MutableCollection<in R>> Sequence<T>.mapIndexedNotNullTo(destination: C, transform: (index: Int, T) -> R?): C {
    return destination.mapIndexedNotNull(transform)\n}

/**
 * Applies the given [transform] function\n * to each element in the original sequence\n * and appends only the non-null results to the given [destination].\n *\n * The operation is _terminal_.\n */
public inline fun <T, R : Any, C : MutableCollection<in R>> Sequence<T>.mapIndexedTo(destination: C, transform: (T) -> R?): C {
    return destination.mapIndexedNotNull(transform)\n}

/**
 * Returns a sequence containing only the non-null results of applying the given [transform] function\n * to each element in the original sequence.\n *\n * The operation is _intermediate_ and _stateless_.\n */
public fun <T, R> Sequence<T>.mapNotNull(transform: (T) -> R?): Sequence<R> {
    return transformingSequence(this, transform, filterNotNull)\n}

/**
 * Applies the given [transform] function\n * to each element in the original sequence\n * and appends only the non-null results to the given [destination].\n *\n * The operation is _terminal_.\n */
public inline fun <T, R : Any, C : MutableCollection<in R>> Sequence<T>.mapNotNullTo(destination: C, transform: (T) -> R?): C {
    return destination.mapNotNull(transform)\n}

/**
 * Applies the given [transform] function\n * to each element of the original sequence\n * and appends the results to the given [destination].\n *\n * The operation is _terminal_.\n */
public inline fun <T, R, C : MutableCollection<in R>> Sequence<T>.mapTo(destination: C, transform: (T) -> R): C {
    return destination.map(transform)\n}

/**
 * Returns a new [MutableSet] containing all distinct elements from the given sequence.\n *\n * The returned set\n */
public fun <T> Sequence<T>.distinct(): Sequence<T> {
    return this.distinctBy\n}

/**
 * Returns a new [MutableSet] containing all distinct elements from the given sequence.\n *\n * The returned set\n */
public fun <T, K> Sequence<T>.distinctBy(selector: (T) -> K): Sequence<T> {
    return this.distinctBy\n}
preserves the element iteration order of the original sequence.

The operation is _terminal_.

*public fun <T> Sequence<T>.toMutableSet(): MutableSet<T> {
    val set = LinkedHashSet<T>()
    for (item in this)
        set.add(item)
    return set
}*

*Returns `true` if all elements match the given [predicate].

*The operation is _terminal_.

*public fun <T> Sequence<T>.all(predicate: (T) -> Boolean): Boolean {
    for (item in this)
        if (!predicate(item)) return false
    return true
}*

*Returns `true` if sequence has at least one element.

*The operation is _terminal_.

*public fun <T> Sequence<T>.any(): Boolean {
    return iterator().hasNext()
}*

*Returns the number of elements in this sequence.

*The operation is _terminal_.

*public fun <T, R> Sequence<T>.fold(initial: R, operation: (acc: R, T) -> R): R {
    var accumulator = initial
    for (element in this)
        accumulator = operation(accumulator, element)
    return accumulator
}*

*Accumulates value starting with [initial] value and applying [operation] from left to right
in current accumulator value and each element.

*The operation is _terminal_.

*public fun <T> Sequence<T>.forEach(action: (T) -> Unit): Unit {
    for (element in this)
        action(element)
}*

*Performs the given [action] on each element.

*The operation is _terminal_.

*public fun <T> Sequence<T>.forEachIndexed(action: (index: Int, T) -> Unit): Unit {
    var index = 0
    for (item in this)
        action(checkIndexOverflow(index++), item)
}*

*Performs the given [action] on each element, providing sequential index with the element.

*The operation is _terminal_.

*public fun <T> Sequence<T>.max(): T? {
    return maxByOrNull(selector)
}*

*Returns the first element yielding the largest value of the given function or `null` if there are no elements.

*The operation is _terminal_.

*
Comparable<R>> Sequence<T>.maxByOrNull(selector: (T) -> R): T? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var maxValue = selector(iterator.next())
    var maxElem = iterator.next()
    do {
        val e = iterator.next()
        val v = selector(e)
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    } while (iterator.hasNext())
    return maxElem
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the sequence. Returns `null` if any of values produced by [selector] function is `NaN`, the returned result is `NaN`. The operation is _terminal_.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <T> Sequence<T>.maxOf(selector: (T) -> Double): Double {
    val iterator = iterator()
    if (!iterator.hasNext()) throw NoSuchElementException()
    var maxValue = selector(iterator.next())
    while (iterator.hasNext()) {
        val v = selector(iterator.next())
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the sequence or `null` if there are no elements. Returns `NaN` if any of values produced by [selector] function is `NaN`, the returned result is `NaN`. The operation is _terminal_.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <T> Sequence<T>.maxOfOrNull(selector: (T) -> Double): Double? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var maxValue = selector(iterator.next())
    while (iterator.hasNext()) {
        val v = selector(iterator.next())
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the sequence or `null` if there are no elements. Returns `NaN` if any of values produced by [selector] function is `NaN`, the returned result is `NaN`. The operation is _terminal_.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <T> Sequence<T>.maxOfOrNull(selector: (T) -> Float): Float? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var maxValue = selector(iterator.next())
    while (iterator.hasNext()) {
        val v = selector(iterator.next())
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the sequence or `null` if there are no elements. Returns `NaN` if any of values produced by [selector] function is `NaN`, the returned result is `NaN`. The operation is _terminal_.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <T, R : Comparable<R>> Sequence<T>.maxOfOrNull(selector: (T) -> R): R? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var maxValue = selector(iterator.next())
    while (iterator.hasNext()) {
        val v = selector(iterator.next())
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue
}
null
var maxValue = selector(iterator.next())
while (iterator.hasNext()) {
    var v = selector(iterator.next())
    if (maxValue < v) {
        maxValue = v
    }
} return maxValue

/**
 * Returns the largest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the sequence.
 */
public inline fun <T, R>
Sequence<T>.maxOf(comparator: Comparator<in R>, selector: (T) -> R): R {
    val iterator = iterator()
    if (!iterator.hasNext()) throw NoSuchElementException()
    var maxValue = selector(iterator.next())
    while (iterator.hasNext()) {
        val v = selector(iterator.next())
        if (comparator.compare(maxValue, v) < 0) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value according to the provided
 * [comparator] among all values produced by [selector] function applied to each element in the sequence or `null` if there are no elements.
 */
public fun <T> Sequence<T>.maxOrNull(): T? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var max = iterator.next()
    while (iterator.hasNext()) {
        val e = iterator.next()
        max = maxOf(max, e)
    }
    return max
}

@Deprecated("Use minOrNull instead.",
ReplaceWith("this.minOrNull()"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun <T> Sequence<T>.min(): T? {
    return minOrNull()
}

@Deprecated("Use minByOrNull instead.",
ReplaceWith("this.minByOrNull(selector)"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun <T>
Sequence<T>.minBy(selector: (T) -> Any): T? {
    val iterator = iterator()
    if (!iterator.hasNext()) throw NoSuchElementException()
    var min = iterator.next()
    while (iterator.hasNext()) {
        val v = iterator.next()
        if (comparator.compare(min, v) < 0) {
            min = v
        }
    }
    return min
}
```kotlin
public inline fun <T, R : Comparable<R>> Sequence<T>.minBy(selector: (T) -> R): T? {
    var minElement = minElement
    do {
        val e = iterator.next()
        val v = selector(e)
        if (minValue > v) {
            minElement = e
            minValue = v
        }
    } while (iterator.hasNext())
    return minElement
}

/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the sequence. \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 * In case of empty sequence, returns the value of `null`.
 * 
 * The operation is _terminal_.
 * 
 * @throws NoSuchElementException if the sequence is empty.
 * 
 * The operation is _terminal_.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <T> Sequence<T>.minOf(selector: (T) -> Double): Double {
    val iterator = iterator()
    if (!iterator.hasNext()) throw NoSuchElementException()
    var minValue = selector(iterator.next())
    while (iterator.hasNext()) {
        val v = selector(iterator.next())
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the sequence or `null` if there are no elements.
 * 
 * The operation is _terminal_.
 * 
 * @throws NoSuchElementException if the sequence is empty.
 * 
 * The operation is _terminal_.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <T> Sequence<T>.minOfOrNull(selector: (T) -> Double): Double? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var minValue = selector(iterator.next())
    while (iterator.hasNext()) {
        val v = selector(iterator.next())
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the sequence or `null` if there are no elements.
 * 
 * The operation is _terminal_.
 * 
 * @throws NoSuchElementException if the sequence is empty.
 * 
 * The operation is _terminal_.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <T> Sequence<T>.minOfOrNull(selector: (T) -> Float): Float? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var minValue = selector(iterator.next())
    while (iterator.hasNext()) {
        val v = selector(iterator.next())
        minValue = minOf(minValue, v)
    }
    return minValue
}
```

operation is _terminal_.

*\@SinceKotlin("1.4.4")\*\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\*\@OverloadResolution

ByLambdaReturnType\*\@on\*\@kotlin.internal.InlineOnly\*\@public inline fun <T, R : Comparable<R>>
Sequence<T>.minOfOrNull(selector: (T) -> R): R? {
    var minValue = selector(Iterator.next())
    while (Iterator.hasNext()) {
        var v = selector(Iterator.next())
        if (minValue > v) minValue = v
    }
    return minValue
}

/**
 * Returns the smallest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the sequence.
 * @throws NoSuchElementException if the sequence is empty.
 * The operation is _terminal_.
 */
*\@SinceKotlin("1.4.4")\*\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\*\@OverloadResolution

ByLambdaReturnType\*\@on\*\@kotlin.internal.InlineOnly\*\@public inline fun <T, R : Comparable<R>>
Sequence<T>.minOfWith(comparator: Comparator<R>, selector: (T) -> R): R {
    var minValue = selector(Iterator.next())
    while (Iterator.hasNext()) {
        var v = selector(Iterator.next())
        if (comparator.compare(minValue, v) > 0) minValue = v
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function applied to each element in the sequence or `null` if there are no elements.
 * The operation is _terminal_.
 */
*\@SinceKotlin("1.4.4")\*\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\*\@OverloadResolution

ByLambdaReturnType\*\@on\*\@kotlin.internal.InlineOnly\*\@public inline fun <T : Comparable<T>>
Sequence<T>.minOrNull(): T? {
    var min = Iterator.next()
    while (Iterator.hasNext()) {
        var e = Iterator.next()
        if (min > e) min = e
    }
    return min
}

@Deprecated("Use minWithOrNull instead.",
ReplaceWith("this.minWithOrNull(comparator)"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun <T> Sequence<T>.minWithOrNull(comparator: Comparator<T>): T? {
    var min = Iterator.next()
    while (Iterator.hasNext()) {
        var e = Iterator.next()
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}

/**
 * Returns `true` if the sequence has no elements.
 * The operation is _terminal_.
 * @sample samples.collections.Collections.Aggregates.none
 */
public inline fun <T> Sequence<T>.none(): Boolean {
    return !Iterator.hasNext()
}

/**
 * Returns `true` if no elements match the given [predicate].
 * The operation is _terminal_.
 * @sample samples.collections.Collections.Aggregates.noneWithPredicate
 */
public inline fun <T> Sequence<T>.none(predicate: (T) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return false
    return true
}

/**
 * Returns a sequence which performs the given [action]
on each element of the original sequence as they pass through it. * The operation is _intermediate_ and _stateless_.

```kotlin
@SinceKotlin("1.1")
public fun <T> Sequence<T>.onEach(action: (T) -> Unit): Sequence<T> {
    return map {
        action(it)
        it
    }
}
```

```kotlin
/**
 * Returns a sequence which performs the given [action] on each element of the original sequence as they pass through it. * @param [action] function that takes the index of an element and the element itself * and performs the action on the element.
 * The operation is _intermediate_ and _stateless_.
 */
@SinceKotlin("1.4")
public fun <T> Sequence<T>.onEachIndexed(action: (Int, T) -> Unit): Sequence<T> {
    return mapIndexed { index, element ->
        action(index, element)
        element
    }
}
```

```kotlin
/**
 * Accumulates value starting with the first element and applying [operation] from left to right * to current accumulator value and each element. * The operation is _terminal_.
 * @param [operation] function that takes current accumulator value and an element,
 * and calculates the next accumulator value.
 */
public inline fun <S, T : S> Sequence<T>.reduce(operation: (acc: S, T) -> S): S {
    val iterator = this.iterator()
    if (!iterator.hasNext())
        throw NoSuchElementException("Empty sequence can't be reduced.")
    var accumulator: S =
    while (iterator.hasNext())
        accumulator = operation(accumulator, iterator.next())
    return accumulator
}
```

```kotlin
/**
 * Accumulates value starting with the first element and applying [operation] from left to right * to current accumulator value and each element with its index in the original sequence. * The operation is _terminal_.
 * @param [operation] function that takes the index of an element, current accumulator value and the element itself,
 * and calculates the next accumulator value.
 */
@SinceKotlin("1.4")
public inline fun <S, T : S> Sequence<T>.reduceIndexed(operation: (index: Int, acc: S, T) -> S): S {
    val iterator = this.iterator()
    if (!iterator.hasNext())
        throw NoSuchElementException("Empty sequence can't be reduced.")
    var index = 1
    var accumulator: S = iterator.next()
    while (iterator.hasNext())
        accumulator = operation(checkIndexOutOfBoundsException(index++), accumulator, iterator.next())
    return accumulator
}
```

```kotlin
/**
 * Returns a sequence containing successive accumulation values generated by applying [operation] from left to right * to current accumulator value and each element with its index in the original sequence. * The [initial] value should also be immutable (or should not be mutated) * as it may be passed to [operation] function
 * @param [operation] function that takes current accumulator value and an element,
 * and calculates the next accumulator value.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public inline fun <S, T : S> Sequence<T>.reduceOrNull(operation: (acc: S, T) -> S): S? {
    val iterator = this.iterator()
    if (!iterator.hasNext())
        return null
    var accumulator: S = iterator.next()
    while (iterator.hasNext())
        accumulator = operation(accumulator, iterator.next())
    return accumulator
}
```
later because of sequence's lazy nature.

* @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value. The operation is _intermediate_ and _stateless_.
* @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value. The operation is _intermediate_ and _stateless_.
* @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value. The operation is _intermediate_ and _stateless_.

 */
@SinceKotlin("1.4")
public fun <T, R> Sequence<T>.runningFold(initial: R, operation: (acc: R, T) -> R): Sequence<R> {
    return sequence {
        yield(initial)
        var accumulator = initial
        for (element in this@runningFold) {
            accumulator = operation(accumulator, element)
            yield(accumulator)
        }
    }
}

/**
 * Returns a sequence containing successive accumulation values generated by applying [operation] from left to right to each element, its index in the original sequence and current accumulator value that starts with [initial] value.
 * Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting sequence.
 * The [initial] value should also be immutable (or should not be mutated) as it may be passed to [operation] function later because of sequence's lazy nature.
 * @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value. The operation is _intermediate_ and _stateless_.
 * @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value. The operation is _intermediate_ and _stateless_.
 * @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value. The operation is _intermediate_ and _stateless_.
 * @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value. The operation is _intermediate_ and _stateless_.

 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public fun <T, R> Sequence<T>.runningReduceIndexed(operation: (index: Int, acc: R, T) -> R): Sequence<R> {
    return sequence {
        val iterator = iterator()
        if (iterator.hasNext()) {
            var accumulator: R = iterator.next()
            var index = 0
            while (iterator.hasNext()) {
                accumulator = operation(checkIndexOverflow(index++), accumulator, iterator.next())
                yield(accumulator)
            }
        }
    }
}

/**
 * Returns a sequence containing successive accumulation values generated by applying [operation] from left to right to each element, its index in the original sequence and current accumulator value that starts with the first element of this sequence.
 * Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting sequence.
 * The [initial] value should also be immutable (or should not be mutated) as it may be passed to [operation] function later because of sequence's lazy nature.
 * The [initial] value should also be immutable (or should not be mutated) as it may be passed to [operation] function later because of sequence's lazy nature.
 * The [initial] value should also be immutable (or should not be mutated) as it may be passed to [operation] function later because of sequence's lazy nature.
 * The [initial] value should also be immutable (or should not be mutated) as it may be passed to [operation] function later because of sequence's lazy nature.

 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public fun <T, R> Sequence<T>.runningReduceIndexed(operation: (index: Int, acc: R, T) -> R): Sequence<R> {
    return sequence {
        val iterator = iterator()
        if (iterator.hasNext()) {
            var accumulator: R = iterator.next()
            var index = 0
            while (iterator.hasNext()) {
                accumulator = operation(checkIndexOverflow(index++), accumulator, iterator.next())
                yield(accumulator)
            }
        }
    }
}

/**
 * Returns a sequence containing successive accumulation values generated by applying [operation] from left to right to each element, its index in the original sequence and current accumulator value that starts with [initial] value.
 * Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting sequence.

 */
@SinceKotlin("1.4")
public fun <T, R> Sequence<T>.scan() {
    return sequence {
        val iterator = iterator()
        if (iterator.hasNext()) {
            var accumulator: R = iterator.next()
            yield(accumulator)
            var index = 0
            while (iterator.hasNext()) {
                accumulator = operation(checkIndexOverflow(index++), accumulator, iterator.next())
                yield(accumulator)
            }
        }
    }
}
Sequence<T>.scan(initial: R, operation: (acc: R, T) -> R): Sequence<R> {
    return runningFold(initial, operation)
}

/**
 * Returns a sequence containing successive accumulation values generated by applying
 * [operation] from left to right to each element, its index in the original sequence and current accumulator value that
 * starts with [initial] value.
 * Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting sequence.
 * The [initial] value should also be immutable (or should not be mutated) as it may be passed to [operation] function later because of sequence’s lazy nature.
 * The operation is _intermediate_ and _stateless_.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public fun <T, R> Sequence<T>.scanIndexed(initial: R, operation: (index: Int, acc: R, T) -> R): Sequence<R> {
    return runningFoldIndexed(initial, operation)
}

@Deprecated("Use runningReduce instead.",
ReplaceWith("runningReduce(operation)"), level =
DeprecationLevel.ERROR)
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public fun <S, T : S> Sequence<T>.scanReduce(operation: (acc: S, T) -> S): Sequence<S> {
    return runningReduce(operation)
}

@Deprecated("Use runningReduceIndexed instead.",
ReplaceWith("runningReduceIndexed(operation)"), level =
DeprecationLevel.ERROR)
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public fun <S, T : S> Sequence<T>.scanReduceIndexed(operation: (index: Int, acc: S, T) -> S): Sequence<S> {
    return runningReduceIndexed(operation)
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the sequence.
 * The operation is _terminal_.
 */
public inline fun <T> Sequence<T>.sumBy(selector: (T) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the sequence.
 * The operation is _terminal_.
 */
public inline fun <T> Sequence<T>.sumByDouble(selector: (T) -> Double): Double {
    var sum: Double = 0.0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the sequence.
 * The operation is _terminal_.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@kotlin.jvm.JvmName("sumOfDouble")
@kotlin.internal.InlineOnly
public inline fun <T> Sequence<T>.sumOf(selector: (T) -> Double): Double {
    var sum: Double = 0.toDouble()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the sequence.
 * The operation is _terminal_.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@kotlin.jvm.JvmName("sumOfInt")
@kotlin.internal.InlineOnly
public inline fun <T> Sequence<T>.sumOf(selector: (T) -> Int): Int {
    var sum: Int = 0.toInt()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the sequence.
 * The operation is _terminal_.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@kotlin.jvm.JvmName("sumOfLong")
@kotlin.internal.InlineOnly
public inline fun <T> Sequence<T>.sumOf(selector: (T) -> Long): Long {
    var sum: Long = 0.toLong()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the sequence.
 * The operation is _terminal_.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@kotlin.jvm.JvmName("sumOfUInt")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <T> Sequence<T>.sumOf(selector: (T) -> UInt): UInt {
    var sum: UInt = 0.toUInt()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}
public inline fun <T> Sequence<T>.sumOf(selector: (T) -> ULong): ULong {
    var sum: ULong = 0.toULong()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns an original collection containing all the non-null elements, throwing an [IllegalArgumentException] if there are any `null` elements.\n * The operation is _intermediate_ and _stateless_.\n */
public fun <T> Sequence<T>.requireNoNulls(): Sequence<T> {
    return map { it ?: throw IllegalArgumentException("null element found in $this.") }
}

/**
 * Splits this sequence into a sequence of lists each not exceeding the given [size].\n * The last list in the resulting sequence may have fewer elements than the given [size].\n * @param size the number of elements to take in each list, must be positive and can be greater than the number of elements in this sequence.\n * @param output a public inline function \n */
public fun <T> Sequence<T>.chunked(size: Int): Sequence<List<T>> {
    return windowed(size, size, partialWindows = true)
}

/**
 * Splits this sequence into several lists each not exceeding the given [size] and applies the [transform] function to each.\n * The last list may have fewer elements than the given [size].\n * @param transform a public inline function \n */
public fun <T, R> Sequence<T>.chunked(size: Int, transform: (List<T>) -> R): Sequence<R> {
    return windowed(size, size, partialWindows = true, transform = transform)
}

/**
 * Returns a sequence containing all elements of the original sequence without the first occurrence of the given [element].\n * The operation is _intermediate_ and _stateful_.\n */
public operator fun <T> Sequence<T>.minus(element: T): Sequence<T> {
    return object: Sequence<T> {
        override fun iterator(): Iterator<T> {
            var removed = false
            return this@minus.filter { if (!removed && it == element) { removed = true; false } else true }.iterator()
        }
    }
}

/**
 * Returns a sequence containing all elements of original sequence except the elements contained in the given [elements] collection.\n * Note that the source sequence and the collection being subtracted are iterated only when an `iterator` is requested from\n * the resulting sequence. Changing any of them between successive calls to `iterator` may affect the result.\n * The [elements] collection may be converted to a [HashSet] to speed up the operation, thus the elements are required to have\n * a correct and stable implementation of `hashCode()` that doesn't change between successive invocations.\n * @param output a public inline function \n */
public operator fun <T> Sequence<T>.minus(elements: Iterable<T>): Sequence<T> {
    return object: Sequence<T> {
        override fun iterator(): Iterator<T> {
            val other = elements.convertToSetForSetOperation()
            if (other.isEmpty())
                return this@minus.iterator()
            else
                return this@minus.filterNot { it in other }.iterator()
        }
    }
}

/**
 * Returns a sequence containing all elements of original sequence except the elements contained in the given [elements] collection.\n * Note that the source sequence and the collection being subtracted are iterated only when an `iterator` is requested from\n * the resulting sequence. Changing any of them between successive calls to `iterator` may affect the result.\n * The [elements] collection may be converted to a [HashSet] to speed up the operation, thus the elements are required to have\n * a correct and stable implementation of `hashCode()` that doesn't change between successive invocations.\n * @param output a public inline function \n */
public operator fun <T> Sequence<T>.minus(elements: Array<out T>): Sequence<T> {
    return object: Sequence<T> {
        override fun iterator(): Iterator<T> {
            val other = elements.toHashSet()
            return this@minus.filterNot { it in other }.iterator()
        }
    }
}

/**
 * Returns a sequence containing all elements of original sequence except the elements contained in the given [elements] collection.\n * Note that the source sequence and the collection being subtracted are iterated only when an `iterator` is requested from\n * the resulting sequence. Changing any of them between successive calls to `iterator` may affect the result.\n * The [elements] collection may be converted to a [HashSet] to speed up the operation, thus the elements are required to have\n * a correct and stable implementation of `hashCode()` that doesn't change between successive invocations.\n * @param output a public inline function \n */
public operator fun <T> Sequence<T>.minus(elements: Iterable<in T>): Sequence<T> {
    return object: Sequence<T> {
        override fun iterator(): Iterator<T> {
            val other = elements.convertToSetForSetOperation()
            if (other.isEmpty())
                return this@minus.iterator()
            else
                return this@minus.filterNot { it in other }.iterator()
        }
    }
}
The \[elements\] sequence may be converted to a [HashSet] to speed up the operation, thus the elements are required to have a correct and stable implementation of `hashCode()` that doesn't change between successive invocations. The operation is _intermediate_ and _stateless_.

```kotlin
public operator fun <T> Sequence<T>.minus(elements: Sequence<T>): Sequence<T> {
    return object: Sequence<T> {
        override fun iterator(): Iterator<T> {
            val other = elements.toHashSet()
            if (other.isEmpty())
                return this@minus.iterator()
            else
                return this@minus.filterNot { it in other }.iterator()
        }
    }
}

/**
 * Returns a sequence containing all elements of the original sequence without the first occurrence of the given \[element\].
 * 
 * The operation is _intermediate_ and _stateless_.
 */
@kotlin.internal.InlineOnly
public inline fun <T> Sequence<T>.minusElement(element: T): Sequence<T> {
    return minus(element)
}

/**
 * Splits the original sequence into pair of lists, \n * where \*[first] list contains elements for which \[predicate\] yielded \`true\', \* while \*[second] list contains elements for which \[predicate\] yielded \`false\'.
 * The operation is _terminal_.
 * 
 * @sample samples.collections.Sequences.Transformations.partition
 */
public inline fun <T> Sequence<T>.partition(predicate: (T) -> Boolean): Pair<List<T>, List<T>> {
    val first = ArrayList<T>()
    val second = ArrayList<T>()
    for (element in this) {
        if (predicate(element)) {
            first.add(element)
        } else {
            second.add(element)
        }
    }
    return Pair(first, second)
}

/**
 * Returns a sequence containing all elements of the original sequence and then the given \[element\].
 * 
 * The operation is _intermediate_ and _stateless_.
 */
@kotlin.internal.InlineOnly
public inline fun <T> Sequence<T>.plusElement(element: T): Sequence<T> {
    return plus(element)
}

/**
 * Returns a sequence of snapshots of the window of the given \[size\]
 * sliding along this sequence with the given \[step\], where each \[snapshot\] is a list. \n * Several last lists may have fewer elements than the given \[size\].
 * Both \[size\] and \[step\] must be positive and can be greater than the number of elements in this sequence.
 * 
 * @param size the number of elements to take in each window
 * @param step the number of elements to move the window forward by on an each step, by default 1
 * @param partialWindows controls whether or not to keep partial windows in the end if any, \n * by default `false` which means partial windows won't be preserved
 * @param @param sample samples.collections.Sequences.Transformations.takeWindows
 */
@SinceKotlin("1.2")
public fun <T> Sequence<T>.windowed(size: Int, step: Int = 1, partialWindows: Boolean = false): Sequence<Sequence<T>> {
    return windowedSequence(size, step, partialWindows, reuseBuffer = false)
}
```
should not store it or allow it to escape in some way, unless you made a snapshot of it. Several last lists may have fewer elements than the given [size]. Both [size] and [step] must be positive and can be greater than the number of elements in this sequence. Returns a sequence of values built from the elements of ‘this’ sequence and the [other] sequence with the same index. The resulting sequence ends as soon as the shortest input sequence ends.

Both [size] and [step] must be positive and can be greater than the number of elements in this sequence. Returns a sequence of values built from the elements of ‘this’ sequence and the [other] sequence with the same index. The resulting sequence ends as soon as the shortest input sequence ends.

The operation is _intermediate_and_stateless_. The returned sequence is empty if this sequence contains less than two elements. The operation is _intermediate_and_stateless_. The returned sequence is empty if this sequence contains less than two elements.

The operation is _intermediate_and_stateless_. The returned sequence is empty if this sequence contains less than two elements. The operation is _intermediate_and_stateless_. The returned sequence is empty if this sequence contains less than two elements.

The operation is _intermediate_and_stateless_. The returned sequence is empty if this sequence contains less than two elements. The operation is _intermediate_and_stateless_. The returned sequence is empty if this sequence contains less than two elements. The operation is _intermediate_and_stateless_. The returned sequence is empty if this sequence contains less than two elements.
kotlin.ranges.contains\(\)\n\nimport kotlin.ranges.reversed\n\n/**
 * Returns a set containing all elements of the original set except the given \[element\].\n * The returned set preserves the element iteration order of the original set.\n */
public operator fun \(<T> Set\<T\>.minus(element: T): Set\<T\> \{ \n    val result = LinkedHashSet\<T\>(mapCapacity(size))\n    var removed = false\n    return this.filterTo(result) \{ \n        if (!removed && \&\& it == element) \{ removed = true; false \} else true \} \n    \n    return result
\} \n\n/**
 * Returns a set containing all elements of the original set except the elements contained in the given \[elements\] array.\n * The returned set preserves the element iteration order of the original set.\n */
public operator fun \(<T> Set\<T\>.minus(elements: Array<out \T\>): Set\<T\> \{ \n    val result = LinkedHashSet\<T\>(this)\n    return result.removeAll(elements)\n    \n    return result
\} \n\n/**
 * Returns a set containing all elements of the original set except the elements contained in the given \[elements\] collection.\n * The returned set preserves the element iteration order of the original set.\n */
public operator fun \(<T> Set\<T\>.minus(elements: Iterable<\T\>): Set\<T\> \{ \n    val other = elements.convertToSetForSetOperationWith(this)\n    if (other.isEmpty())\n        return this.toSet()\n    if (other is Set)\n        return this.filterNotTo(LinkedHashSet\<T\>()) \{ \n            it in other \n        }\n    val result = LinkedHashSet\<T\>(mapCapacity(this.size + elements.collectionSizeOrNull()!!))\n    result.removeAll(other)\n    return result
\} \n\n/**
 * Returns a set containing all elements of the original set except the given \[element\].\n * The returned set preserves the element iteration order of the original set.\n */
@kotlin.internal.InlineOnly
public inline fun \(<T> Set\<T\>.minusElement(element: T): Set\<T\> \{ \n    return minus(element)\n\} \n\n/**
 * Returns a set containing all elements of the original set and then the given \[element\] if it isn't already in this set.\n * The returned set preserves the element iteration order of the original set.\n */
public operator fun \(<T> Set\<T\>.plus(element: T): Set\<T\> \{ \n    val result = LinkedHashSet\<T\>(mapCapacity(size + 1))\n    result.addAll(this)\n    result.add(element)\n    return result
\} \n\n/**
 * Returns a set containing all elements of the original set and the given \[elements\] array, which aren't already in this set.\n * The returned set preserves the element iteration order of the original set.\n */
public operator fun \(<T> Set\<T\>.plus(elements: Array<out \T\>): Set\<T\> \{ \n    val result = LinkedHashSet\<T\>(mapCapacity(this.size + elements.size))\n    result.addAll(this)\n    result.addAll(elements)\n    return result
\} \n\n/**
 * Returns a set containing all elements of the original set and the given \[elements\] collection, which aren't already in this set.\n * The returned set preserves the element iteration order of the original set.\n */
public operator fun \(<T> Set\<T\>.plus(elements: Iterable<\T\>): Set\<T\> \{ \n    val result = LinkedHashSet\<T\>(mapCapacity(elements.collectionSizeOrNull() ?: this.size * 2))\n    result.addAll(this)\n    result.addAll(elements)\n    return result
\} \n\n/**
 * Returns a set containing all elements of the original set and the given \[elements\] sequence, which aren't already in this set.\n * The returned set preserves the element iteration order of the original set.\n */
public operator fun \(<T> Set\<T\>.plus(elements: Sequence<\T\>): Set\<T\> \{ \n    val result = LinkedHashSet\<T\>(mapCapacity(this.size + 2))\n    result.addAll(this)\n    result.addAll(elements)\n    return result
\}
kotlin.text

// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt
See: https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib

import kotlin.random.*

/**
 * Returns a character at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this char sequence.
 */
public expect fun CharSequence.elementAt(index: Int): Char

/**
 * Returns a character at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this char sequence.
 */
@kotlin.internal.InlineOnly
public inline fun CharSequence.getOrElse(index: Int, defaultValue: (Int) -> Char): Char {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}

/**
 * Returns a character at the given [index] or `null` if the [index] is out of bounds of this char sequence.
 */
@kotlin.internal.InlineOnly
public inline fun CharSequence.getOrNull(index: Int): Char? {
    return if (index >= 0 && index <= lastIndex) get(index) else null
}

/**
 * Returns index of the first character matching the given [predicate], or `null` if no such character was found.
 */
public inline fun CharSequence.indexOfFirst(predicate: (Char) -> Boolean): Int {
    for (index in indices) {
        if (predicate(this[index])) {
            return index
        }
    }
    return -1
}

/**
 * Returns index of the last character matching the given [predicate], or -1 if the char sequence does not contain such character.
 */
public inline fun CharSequence.indexOfLast(predicate: (Char) -> Boolean): Int {
    for (index in indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return index
    }
    return -1
}

/**
 * Returns the first character matching the given [predicate], or `null` if no such character was found.
 */
public inline fun CharSequence.firstOrNull(predicate: (Char) -> Boolean): Char? {
    for (index in indices) {
        val element = this[index]
        if (predicate(element)) return element
    }
    return null
}

/**
 * Returns the last character matching the given [predicate], or `null` if no such character was found.
 */
public inline fun CharSequence.lastOrNull(predicate: (Char) -> Boolean): Char? {
    for (index in indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    return null
}

/**
 * Returns first character.
 * @throws NoSuchElementException if the char sequence is empty.
 */
public fun CharSequence.first(): Char {
    if (isEmpty())
        throw NoSuchElementException("Char sequence is empty.")
    return this[0]
}

/**
 * Returns the first character matching the given [predicate].
 * @throws NoSuchElementException if no such character was found.
 */
public inline fun CharSequence.first(predicate: (Char) -> Boolean): Char {
    for (index in this.indices) {
        val element = this[index]
        if (predicate(element)) return element
    }
    throw NoSuchElementException("Char sequence contains no character matching the predicate.")
}

/**
 * Returns the last character.
 * @throws NoSuchElementException if the char sequence is empty.
 */
public fun CharSequence.last(): Char {
    if (isEmpty())
        throw NoSuchElementException("Char sequence is empty.")
    return this[lastIndex]
}

/**
 * Returns the last character matching the given [predicate].
 * @throws NoSuchElementException if no such character was found.
 */
public inline fun CharSequence.last(predicate: (Char) -> Boolean): Char {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    throw NoSuchElementException("Char sequence contains no character matching the predicate.")
}
empty.

```kotlin
public fun CharSequence.lastOrNull(): Char? {
    return if (isEmpty()) null else this[length - 1]
}

public inline fun CharSequence.lastOrNull(predicate: (Char) -> Boolean): Char? {
    return get(random.nextInt(length))
}

/**
 * Returns the single character, or throws an exception if the char sequence is empty or has more than one character.
 */
public fun CharSequence.single(): Char {
    return when (length) {
        0 -> throw NoSuchElementException("Char sequence is empty.")
        1 -> this[0]
        else -> throw IllegalArgumentException("Char sequence has more than one element.")
    }
}

/**
 * Returns a subsequence of this char sequence with the first [n] characters removed.
 * @throws IllegalArgumentException if [n] is negative.
 */
public fun CharSequence.drop(n: Int): CharSequence {
    return subSequence(n.coerceAtMost(length), length)
}

/**
 * Returns a string with the first [n] characters removed.
 * @throws IllegalArgumentException if [n] is negative.
 */
public fun String.drop(n: Int): String {
    return substring(n.coerceAtMost(length))
}

/**
 * Returns a subsequence of this char sequence containing all characters except last characters that satisfy the given [predicate].
 */
public fun CharSequence.dropLast(n: Int): CharSequence {
    return subSequence(n.coerceAtMost(length), length)
}

/**
 * Returns a string with the last [n] characters removed.
 * @throws IllegalArgumentException if [n] is negative.
 */
public fun String.dropLast(n: Int): String {
    return substring(n.coerceAtMost(length), length)
}
```

---

**Open Source Used in security_automation identity-guest-cs 1453**

* @sample samples.text.Strings.last

* @sample samples.text.Strings.drop

* @sample samples.text.Strings.dropLast

---

**@sample samples.test.Strings.lastOrNull()**

```kotlin
public fun CharSequence.lastOrNull(): Char? {
    return if (isEmpty()) null else this[length - 1]
}
```

**@sample samples.text.Strings.dropOrNonNull()**

```kotlin
public fun CharSequence.dropOrNonNull(predicate: (Char) -> Boolean): Char? {
    return get(random.nextInt(length))
}
```

---

**@sample samples.text.Strings.random()**

```kotlin
public fun Random.random(): Char {
    return random.Random
}
```

**@sample samples.text.Strings.randomOrNull()**

```kotlin
public fun Random.randomOrNull(): Char? {
    return random.Random
}
```

---

**@sample samples.text.Strings.single()**

```kotlin
public fun CharSequence.single(): Char {
    return when (length) {
        0 -> throw NoSuchElementException("Char sequence is empty.")
        1 -> this[0]
        else -> throw IllegalArgumentException("Char sequence has more than one element.")
    }
}
```

**@sample samples.text.Strings.singleOrNull()**

```kotlin
public fun CharSequence.singleOrNull(): Char? {
    return if (length == 1) this[0] else null
}
```

---

**@sample samples.text.Strings.drop()**

```kotlin
public fun String.drop(n: Int): String {
    return substring(n.coerceAtMost(length))
}
```

**@sample samples.text.Strings.dropLast()**

```kotlin
public fun String.dropLast(n: Int): String {
    return substring(n.coerceAtMost(length))
}
```
```kotlin
// Public inline fun CharSequence.dropLastWhile(predicate: (Char) -> Boolean): CharSequence {
//     for (index in lastIndex downTo 0) {
//         if (!predicate(this[index])) return subSequence(0, index + 1)
//     }
//     return ""
// }

/**
 * Returns a string containing all characters except last characters that satisfy the given [predicate].
 * @sample samples.text.Strings.drop
 */
public inline fun String.dropLastWhile(predicate: (Char) -> Boolean): String {
    for (index in lastIndex downTo 0)
        if (!predicate(this[index]))
            return substring(0, index + 1)
    return ""
}

/**
 * Returns a subsequence of this char sequence containing all characters except first characters that satisfy the given [predicate].
 * @sample samples.text.Strings.drop
 */
public inline fun CharSequence.dropWhile(predicate: (Char) -> Boolean): CharSequence {
    for (index in this.indices)
        if (!predicate(this[index]))
            return subSequence(index, length)
    return ""
}

/**
 * Returns a string containing all characters except first characters that satisfy the given [predicate].
 * @sample samples.text.Strings.drop
 */
public inline fun String.dropWhile(predicate: (Char) -> Boolean): String {
    for (index in this.indices)
        if (!predicate(this[index]))
            return substring(index)
    return ""
}

/**
 * Returns a char sequence containing only those characters from the original char sequence that match the given [predicate].
 * @sample samples.text.Strings.filter
 */
public inline fun CharSequence.filter(predicate: (Char) -> Boolean): CharSequence {
    return filterTo(StringBuilder(), predicate)
}

/**
 * Returns a string containing only those characters from the original string that match the given [predicate].
 * @sample samples.text.Strings.filter
 */
public inline fun String.filter(predicate: (Char) -> Boolean): String {
    return filterTo(StringBuilder(), predicate).toString()
}

/**
 * Returns a char sequence containing only those characters from the original char sequence that match the given [predicate].
 * @param [predicate] function that takes the index of a character and the character itself
 * and returns the result of predicate evaluation on the character.
 * @sample samples.collections.Collections.Filtering.filterIndexed
 */
public inline fun CharSequence.filterIndexed(predicate: (index: Int, Char) -> Boolean): CharSequence {
    return filterIndexedTo(StringBuilder(), predicate)
}

/**
 * Returns a string containing only those characters from the original string that match the given [predicate].
 * @param [predicate] function that takes the index of a character and the character itself
 * and returns the result of predicate evaluation on the character.
 * @sample samples.collections.Collections.Filtering.filterTo
 */
public inline fun String.filterIndexed(predicate: (index: Int, Char) -> Boolean): String {
    return filterIndexedTo(StringBuilder(), predicate).toString()
}

/**
 * Appends all characters matching the given [predicate] to the given [destination].
 * @param [predicate] function that takes the index of a character and the character itself
 * and returns the result of predicate evaluation on the character.
 * @sample samples.collections.Collections.Filtering.filterNot
 */
public inline fun CharSequence.filterNot(predicate: (Char) -> Boolean):(CharSequence) {
    return filterNotTo(StringBuilder(), predicate)
}

/**
 * Appends all characters not matching the given [predicate] to the given [destination].
 * @param [predicate] function that takes the index of a character and the character itself
 * and returns the result of predicate evaluation on the character.
 * @sample samples.collections.Collections.Filtering.filterNotTo
 */
public inline fun String.filterNot(predicate: (Char) -> Boolean): String {
    return filterNotTo(StringBuilder(), predicate).toString()
}
```

---

Open Source Used In security_automation identity-guest-cs 1454
(indices.isEmpty()) return "" * Returns a string containing characters of the original string at the specified range of [indices].

public fun String.slice(indices: IntRange): String {
    if (indices.isEmpty()) return "" * Returns a char sequence containing characters of the original char sequence at specified [indices].

    return subSequence(indices)
}

/**
 * Returns a string containing characters of the original string at specified [indices].
 */
public fun CharSequence.slice(indices: IntRange): String {
    val size = indices.collectionSizeOrDefault(10)
    if (size == 0) return "" * Returns a string containing characters of the original string at specified [indices].

    return subSequence(0, n.coerceAtMost(length))
}

/**
 * Returns a char sequence containing characters of the original char sequence at specified [indices].
 */
public fun CharSequence.sliceLast(n: Int): CharSequence {
    val length = length
    return subSequence(length - n.coerceAtMost(length), length)
}

/**
 * Returns a string containing characters of the original string at specified [indices].
 */
@kotlin.internal.InlineOnly
public inline fun String.slice(indices: Iterable<Int>): String {
    return (this as CharSequence).slice(indices).toString()
}

/**
 * Returns a string containing characters of the original string at specified [indices].
 */
@kotlin.internal.InlineOnly
public inline fun String.sliceLast(n: Int): String {
    val length = length
    return substring(length - n.coerceAtMost(length))
}

/**
 * Returns a subsequence of this char sequence containing the first [n] characters from this char sequence, or the entire char sequence if this char sequence is shorter.

    @throws IllegalArgumentException if [n] is negative.

    * @sample samples.text.Strings.take
 */
public fun CharSequence.take(n: Int): CharSequence {
    require(n >= 0) { "Requested character count $n is less than zero." }
    return subSequence(0, n.coerceAtMost(length))
}

/**
 * Returns a string containing characters of the original string at specified [indices].
 */
@kotlin.internal.InlineOnly
public inline fun String.take(n: Int): String {
    require(n >= 0) { "Requested character count $n is less than zero." }
    return substring(0, n.coerceAtMost(length))
}

/**
 * Returns a subsequence of this char sequence containing last characters that satisfy the given [predicate].

    @sample samples.text.Strings.take
 */
public inline fun CharSequence.takeLastWhile(predicate: (Char) -> Boolean): CharSequence {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return subSequence(index + 1)
        }
    }
    return subSequence(0, length)
}

/**
 * Returns a string containing characters of the original string at specified [indices].
 */
@kotlin.internal.InlineOnly
public inline fun String.takeLastWhile(predicate: (Char) -> Boolean): String {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return substring(index + 1)
        }
    }
    return this
}

/**
 * Returns a subsequence of this char sequence containing first characters that satisfy the given [predicate].

    @sample samples.text.Strings.take
 */
public inline fun String.takeWhile(predicate: (Char) -> Boolean): String {
    for (index in 0 until length)
        if (!predicate(get(index))) {
            return substring(0, index)
        }
    return substring(0, length)
}

public fun CharSequence.reversed(): CharSequence {
    return StringBuilder(this).reverse()
}

/**
 * Returns a string with characters in reversed order.

    * @sample samples.text.Strings.reverse
 */
@kotlin.internal.InlineOnly
public inline fun String.reversed(): String {
    return (this as CharSequence).reversed().toString()
}

/**
 * Returns a map containing key-value pairs provided by [transform] function * applied to characters of the given char sequence.

    * @sample samples.text.Strings.transform
 */
public fun <K, V> CharSequence.associate(transform: (Char) -> Pair<K, V>): Map<K, V> {
    val capacity = mapCapacity(length).coerceAtLeast(16)
    return associateTo(LinkedHashMap<K, V>(capacity),
    for (i in indices) { CharSequence(result.append(get(i)))
    } return result
}
transf

\n
** Returns a [Map] containing the characters from the given char sequence indexed by the key
* returned from [keySelector] function applied to each character. * * * If any two characters would have the same key returned by [keySelector] the last one gets added to the map. * * * The returned map preserves the entry iteration order of the original char sequence. * * * * @sample samples.text.Strings.associateBy

@public inline fun <K> CharSequence.associateBy(keySelector: (Char) -> K): Map<K, Char> { val capacity = mapCapacity(length).coerceAtLeast(16) return associateByTo(LinkedHashMap<K, Char>(capacity), keySelector) } * * * * Returns a [Map] containing the values provided by [valueTransform] and indexed by [keySelector] functions applied to characters of the given char sequence. * * * * If any two characters would have the same key returned by [keySelector] the last one gets added to the map. * * * The returned map preserves the entry iteration order of the original char sequence. * * * * @sample samples.text.Strings.associateByWith

@public inline fun <K, V> CharSequence.associateByWith(valueSelector: (Char) -> V): Map<K, V> { val capacity = mapCapacity(length).coerceAtLeast(16) return associateByTo(LinkedHashMap<K, V>(capacity), keySelector, valueTransform) } * * * * Returns a [Map] where keys are characters from the given char sequence and values are produced by the [valueSelector] function applied to each character of the given char sequence. * * * * @sample samples.text.Strings.associateWith

@SinceKotlin("1.3") @public inline fun <V> CharSequence.associateWith(valueSelector: (Char) -> V): HashSet<Char> {
    return toCollection(HashSet<Char>(mapCapacity(length.coerceAtMost(128)).coerceAtLeast(16)))
}
Returns a [List] containing all characters.

*\n
**\n
public fun CharSequence.toList(): List<Char> {
    return when
        (length) {
            0 -> emptyList()
            1 -> listOf(this[0])
            else -> this.toMutableList()
        }
}

/**
 * Returns a [Set] of all characters.
 * 
 * The returned set preserves the element iteration order of the original char sequence.
 * */
public fun CharSequence.toSet(): Set<Char> {
    return when
        (length) {
            0 -> emptySet()
            1 -> setOf(this[0])
            else -> toCollection(LinkedHashSet<Char>(mapCapacity(length.coerceAtMost(128))))
        }
}

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each character of original char sequence.
 * 
 * @sample samples.collections.Transformations.flatten
 */
public inline fun <R>
    CharSequence.flatMap(transform: (Char) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(),
        transform)
}

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each character
 * and its index in the original char sequence.
 * 
 * @sample samples.collections.Transformations.flattenIndexed
 */
@SinceKotlin("1.4")
@OptIn(kotlin.internal.InlineOnly::class)
@kotlin.experimental.ExperimentalTypeInference
@kotlin.jvm.JvmName("flatMapIndexed")
public inline fun <R>
    CharSequence.flatMapIndexed(transform: (index: Int, Char) -> Iterable<R>): List<R> {
    return flatMapIndexedTo(ArrayList<R>(),
        transform)
}

/**
 * Groups characters of the original char sequence by
 * the key returned by the given [keySelector] function
 * applied to each character and puts to the [destination] map each group key associated with a list of corresponding characters.
 * 
 * @return The [destination] map.
 * 
 * @sample samples.collections.Transformations.groupBy
 */
public inline fun <K, M : MutableMap<in K, MutableList<Char>>>
    CharSequence.groupByTo(destination: M,
        keySelector: (Char) -> K): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<Char>() }
        list.add(element)
    }
    return destination
}

/**
 * Groups values returned by the [valueTransform] function applied to each character of the original char sequence
 * by the key returned by the given [keySelector] function applied to each character and returns a map where each group key is associated with a list of corresponding values.
 * 
 * @return The map.
 * 
 * @sample samples.collections.Transformations.groupByKeysAndValues
 */
public inline fun <K, V>
    CharSequence.groupBy(keySelector: (Char) -> K, valueTransform: (Char) -> V): Map<K, List<V>> {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<Char>() }
        list.add(element)
    }
    return destination
}
Groups values returned by the [valueTransform] function applied to each character of the original char sequence\n* by the key returned by the given [keySelector] function applied to the character\n* and puts to the [destination] map each group key associated with a list of corresponding values.\n* \n* @return The [destination] map.\n* @sample samples.collections.Collections.Transformations.groupByKeysAndValues
*\n*public inline fun <K, V, M : MutableMap<K, MutableList<V>>> CharSequence.groupByTo(destination: M, keySelector: (Char) -> K, valueTransform: (Char) -> V): M {\nfor (element in this) {\n    val key = keySelector(element)\n    val list = destination.getOrPut(key) { ArrayList<V>() }\n    list.add(valueTransform(element))\n} \nreturn destination\n*\n* Creates a [Grouping] source from a char sequence to be used later with one of group-and-fold operations\n* using the specified [keySelector] function to extract a key from each character.\n* @sample samples.collections.Grouping.groupingByEach\n*\n*CharSequence.groupingBy(crossinline keySelector: (Char) -> K): Grouping<Char, K> {\n    return object : Grouping<Char, K> {\n        override fun sourceIterator(): Iterator<Char> = this@groupingBy.iterator()\n        override fun keyOf(element: Char): K = keySelector(element)\n    }\n*\n*\n* Returns a list containing the results of applying the given [transform] function\n* to each character in the original char sequence.\n* \n* @sample samples.text.Strings.map\n*\n*public inline fun <R> CharSequence.map(transform: (Char) -> R): List<R> {\n    return mapTo(ArrayList<R>(length), transform)\n*\n* Returns a list containing the results of applying the given [transform] function\n* to each character and its index in the original char sequence.\n* \n* @param [transform] function that takes the index of a character and the character itself\n* and returns the result of the transform applied to the character.\n*\n*public inline fun <R> CharSequence.mapIndexed(transform: (index: Int, Char) -> R): List<R> {\n    return mapIndexedTo(ArrayList<R>(length), transform)\n*\n* Returns a list containing only the non-null results of applying the given [transform] function\n* to each character and its index in the original char sequence.\n* \n* @param [transform] function that takes the index of a character and the character itself\n* and returns the result of the transform applied to the character.\n*\n*public inline fun <R : Any> CharSequence.mapIndexedNotNull(transform: (index: Int, Char) -> R?): List<R> {\n    return mapIndexedNotNullTo(ArrayList<R>(), transform)\n*\n* Applies the given [transform] function to each character and its index in the original char sequence\n* and appends only the non-null results to the given [destination].\n* \n* @param [transform] function that takes the index of a character and the character itself\n* and returns the result of the transform applied to the character.\n*\n*public inline fun <R : Any, C : MutableCollection<in R>> CharSequence.mapIndexedNotNullTo(destination: C, transform: (index: Int, Char) -> R?): List<R> {\n    return mapIndexedNotNullTo(destination, transform)\n*\n* Applies the given [transform] function to each character and its index in the original char sequence\n* and appends the results to the given [destination].\n* \n* @param [transform] function that takes the index of a character and the character itself\n* and returns the result of the transform applied to the character.\n*\n*public inline fun <R : Any> CharSequence.mapNotNull(transform: (Char) -> R?): List<R> {\n    return mapNotNullTo(ArrayList<R>(), transform)\n*\n* Applies the given [transform] function to each character in the original char sequence\n* and appends only the non-null results to the given [destination].\n* \n* @param [transform] function that takes the index of a character and the character itself\n* and returns the result of the transform applied to the character.\n*\n*public inline fun <R, C : MutableCollection<in R>> CharSequence.mapNotNullTo(destination: C, transform: (Char) -> R?): List<R> {\n    var index = 0\n    for (element in this) {\n        transform(index++, element)\n        destination.add(index, element)\n    }\n    return destination\n*\n* Applies the given [transform] function to each character of the original char sequence\n* and appends the results to the given [destination].\n* \n* @param [transform] function that takes the index of a character and the character itself\n* and returns the result of the transform applied to the character.\n*\n*public inline fun <R : Any> CharSequence.mapNotNullTo(destination: R) {\n    return mapNotNullTo(destination, transform)\n*\n* Applies the given [transform] function to each character of the original char sequence\n* and appends the results to the given [destination].\n* \n* @param [transform] function that takes the index of a character and the character itself\n* and returns the result of the transform applied to the character.\n*\n*public inline fun <R : Any, C : MutableCollection<in R>> CharSequence.mapNotNullTo(destination: C, transform: (Char) -> R?): List<R> {\n    var index = 0\n    for (element in this) {\n        transform(index++, element)\n        destination.add(index, element)\n    }\n    return destination\n*\n* Applies the given [transform] function to each character of the original char sequence\n* and appends the results to the given [destination].\n* \n* @param [transform] function that takes the index of a character and the character itself\n* and returns the result of the transform applied to the character.\n*\n*public inline fun <R : Any> CharSequence.mapNotNullTo(destination: R) {\n    return mapNotNullTo(destination, transform)\n*\n* Applies the given [transform] function to each character of the original char sequence\n* and appends the results to the given [destination].
CharSequence.withIndex(): Iterable<IndexedValue<Char>> { in return IndexingIterable { iterator() } } in

samples.collections.Collections.Aggregates.allCharSequence

/**
 * Returns `true` if all characters match the given [predicate].
 * @sample samples.collections.Collections.Aggregates.all
 */
public inline fun CharSequence.all(predicate: (Char) -> Boolean): Boolean {
    return !isNotEmpty() && all { predicate(it) }
}

/**
 * Returns the length of this char sequence.
 */
public fun CharSequence.count(): Int {
    return length
}

/**
 * Returns the number of characters matching the given [predicate].
 */
public inline fun CharSequence.count(predicate: (Char) -> Boolean): Int {
    var count = 0
    for (element in this) if (predicate(element)) ++count
    return count
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right
 * to current accumulator value and each character.
 */
public inline fun <R> CharSequence.fold(initial: R, operation: (acc: R, Char) -> R): R {
    var accumulator = initial
    for (element in this) accumulator = operation(accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from right to left
 * to current accumulator value and each character.
 */
public inline fun <R> CharSequence.foldRight(initial: R, operation: (Char, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Performs the given [action] on each character.
 */
public inline fun CharSequence.forEach(action: (Char) -> Unit): Unit {
    for (element in this) action(element)
}

/**
 * Performs the given [action] on each character, providing sequential index with the character.
 */
public inline fun CharSequence.forEachIndexed(action: (index: Int, Char) -> Unit): Unit {
    var index = 0
    for (item in this) action(index++, item)
}

@Deprecated("Use maxOrNull instead.", ReplaceWith("this.maxOrNull()"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun CharSequence.max(): Char? {
    return maxOrNull()
}

@Deprecated("Use maxByOrNull instead.", ReplaceWith("this.maxByOrNull(selector)"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public inline fun <R : Comparable<R>> CharSequence.maxBy(selector: (Char) -> R): Char? {
    return maxByOrNull(selector)
}
@SinceKotlin("1.4")
public inline fun <R : Comparable<R>> CharSequence.maxByOrNull(selector: (Char) -> R): Char? {
    if (isEmpty()) return null
    var maxElem = this[0]
    val lastIndexOf = this.lastIndexOf
    if (lastIndexOf == 0) return maxElem
    val valueArr = selector(this[0])
    for (i in 0 until lastIndex) {
        val char = this[i]
        val value = selector(char)
        if (maxValue < value) {
            maxElem = char
            maxValue = value
        }
    }
    return maxElem
}

/**
 * Returns the largest value among all values produced by [selector] function applied to each character in the char sequence.
 * @throws NoSuchElementException if the char sequence is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
public inline fun CharSequence.maxOf(selector: (Char) -> Double): Double = this

/**
 * Returns the largest value among all values produced by [selector] function applied to each character in the char sequence.
 * 
 * @throws NoSuchElementException if the char sequence is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
public inline fun CharSequence.maxOf(selector: (Char) -> Float): Float = this

/**
 * Returns the largest value among all values produced by [selector] function applied to each character in the char sequence.
 * 
 * @throws NoSuchElementException if the char sequence is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
public inline fun <R : Comparable<R>> CharSequence.maxOf(selector: (Char) -> R): R = this

/**
 * Returns the largest value among all values produced by [selector] function applied to each character in the char sequence.
 * 
 * @throws NoSuchElementException if the char sequence is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
public inline fun CharSequence.maxOfOrNull(selector: (Char) -> Double): Double? = this

/**
 * Returns the largest value among all values produced by [selector] function applied to each character in the char sequence.
 * 
 * @throws NoSuchElementException if the char sequence is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
public inline fun CharSequence.maxOfOrNull(selector: (Char) -> Float): Float? = this

/**
 * Returns the largest value according to the provided [comparator] among all values produced by [selector] function applied to each character in the char sequence.
 * 
 * @throws NoSuchElementException if the char sequence is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
public inline fun <R : Comparable<R>> CharSequence.maxOfOrNull(selector: (Char) -> R): R? = this
ByLambdaReturnType

public inline fun <R> CharSequence.maxOfWith(comparator: Comparator<in R>, selector: (Char) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(maxValue, v) < 0) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value according to the provided [comparator] among all values produced by [selector] function applied to each character in the char sequence or `null` if there are no characters.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType

public inline fun <R> CharSequence.maxOfWithOrNull(comparator: Comparator<in R>, selector: (Char) -> R): R? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(maxValue, v) < 0) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest character or `null` if there are no characters.
 */
@SinceKotlin("1.4")
public fun CharSequence.maxOrNull(): Char? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (max < e) max = e
    }
    return max
}

@Deprecated("Use maxWithOrNull instead.", ReplaceWith("this.maxWithOrNull(comparator)"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun CharSequence.maxWith(comparator: Comparator<Char>): Char? {
    return maxWithOrNull(comparator)
}

/**
 * Returns the first character having the largest value according to the provided [comparator] or `null` if there are no characters.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType

public inline fun <R : Comparable<R>> CharSequence.minBy(selector: (Char) -> R): Char? {
    if (isEmpty()) return null
    var minElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return minElem
    var minValue = selector(minElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (minValue > v) {
            minElem = e
            minValue = v
        }
    }
    return minElem
}

/**
 * Returns the smallest value among all values produced by [selector] function applied to each character in the char sequence.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 * @throws NoSuchElementException if the char sequence is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType

public inline fun <R : Comparable<R>> CharSequence.minByOrNull(selector: (Char) -> R): Char? {
    if (isEmpty()) throw NoSuchElementException()
    var minElem = selector(this[0])
    for (i in 1..lastIndex) {
        val e = selector(this[i])
        if (minElem == e) min = e
    }
    return min
}

/**
 * Returns the smallest value among all values produced by [selector] function applied to each character in the char sequence.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 * @throws NoSuchElementException if the char sequence is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType

public inline fun CharSequence.min(): Char? {
    return minOrNull()
}

@Deprecated("Use minByOrNull instead.", ReplaceWith("this.minByOrNull(selector)"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public inline fun <R : Comparable<R>> CharSequence.minByOrNull(selector: (Char) -> R): Char? {
    if (isEmpty()) throw NoSuchElementException()
    var min = minOrNull()
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = minOrNull()
        if (min > v) {
            min = v
        }
    }
    return min
}

/**
 * Returns the smallest value among all values produced by [selector] function applied to each character in the char sequence.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 * @throws NoSuchElementException if the char sequence is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType

public inline fun <R : Comparable<R>> CharSequence.minBy(selector: (Char) -> R): Char? {
    if (isEmpty()) throw NoSuchElementException()
    var min = minOrNull()
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = minOrNull()
        if (min > v) {
            min = v
        }
    }
    return min
}
ByLambdaReturnType
@kotlin.internal.InlineOnly
@public inline fun <R : Comparable<R>>
CharSequence.minOf(selector: (Char) -> R): R {
    if (isNotEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function * applied to each character in the char sequence or `null` if there are no characters.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
@public inline fun <R : Comparable<R>>
CharSequence.minOfOrNull(selector: (Char) -> R): R? {
    if (isNotEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function * applied to each character in the char sequence or `null` if there are no characters.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
@public inline fun CharSequence.minOfOrNull(selector: (Char) -> Float): Float? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function * applied to each character in the char sequence or `null` if there are no characters.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
@public inline fun <R : Comparable<R>>
CharSequence.minOfOrNull(selector: (Char) -> R): R? {
    if (isNotEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value according to the provided [comparator] among all values produced by [selector] function applied to each character in the char sequence.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
@public inline fun <R> CharSequence.minOfWith(comparator: Comparator<in R>, selector: (Char) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(minValue, v) > 0) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value according to the provided [comparator] among all values produced by [selector] function applied to each character in the char sequence or `null` if there are no characters.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
@public inline fun <R> CharSequence.minOfWithOrNull(comparator: Comparator<in R>, selector: (Char) -> R): R? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(minValue, v) > 0) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest character or `null` if there are no characters.
 */
@SinceKotlin("1.4")
public fun CharSequence.minOrNull(): Char? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (min > e) min = e
    }
    return min
}

@Deprecated("Use minWithOrNull instead.", ReplaceWith("this.minWithOrNull(comparator)"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun CharSequence.minWithOrNull(comparator: Comparator<Char>): Char? {
    if (isNotEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (min > e) min = e
    }
    return min
}

@Deprecated("Use minWithOrNull instead.", ReplaceWith("this.minWithOrNull(comparator)"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun CharSequence.minWithOrNull(comparator: Comparator<Char>): Char? {
    if (isNotEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}

@Deprecated("Use minWithOrNull instead.", ReplaceWith("this.minWithOrNull(comparator)"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun CharSequence.minWithOrNull(comparator: Comparator<Char>): Char? {
    if (isNotEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}

@Deprecated("Use minWithOrNull instead.", ReplaceWith("this.minWithOrNull(comparator)"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun CharSequence.minWithOrNull(comparator: Comparator<Char>): Char? {
    if (isNotEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}
samples.collections.Collections.Aggregates.none

@public fun CharSequence.none(): Boolean { 
    return isEmpty()
} *

@public inline fun CharSequence.none(predicate: (Char) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return false
    return true
} *

/**
 * Performs the given [action] on each character and returns the char sequence itself afterwards.
 */
@public fun @SinceKotlin("1.1.0") public inline fun S.onEach(action: (Char) -> Unit): S { 
    for (element in this) if (action(element)) return
    return this
} *

/**
 * Performs the given [action] on each character, providing sequential index with the character,
 * and returns the char sequence itself afterwards.
 */
@public @param [action] function that takes the index of a character and the character itself,
 * and performs the action on the character.
@public @SinceKotlin("1.4") public inline fun S.onEachIndexed(operation: (index: Int, Char) -> Unit): S { 
    return operation(0, this[0])
    for (index in 1..lastIndex) {
        return operation(index, this[index])
    }
    return this
} *

/**
 * Performs the given [operation] from left to right on each character and returns the accumulator value.
 */
@public inline fun CharSequence.reduce(operation: (acc: Char, Char) -> Char): Char { 
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
} *
samples.collections.Collections.Aggregates.reduceRight

```kotlin
public inline fun
CharSequence.reduceRight(operation: (Char, acc: Char) -> Char): Char {

    if (isEmpty()) throw UnsupportedOperationException("Empty char sequence can't be reduced.")

    var accumulator = get(index-1)
    while (index >= 0) {
        accumulator = operation(get(--index), accumulator)
    }
    return accumulator
}
```

```kotlin
/**
 * Accumulates value starting with the last character and applying [operation] from right to left
 * to each character with its index in the original char sequence and current accumulator value.
 * Throws an exception if this char sequence is empty. If the char sequence can be empty in an expected way,
 * please use [reduceRightIndexedOrNull] instead. It returns `null` when its receiver is empty.

 * @param [operation] function that takes the index of a character, the character itself and current accumulator value,
 * and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduceRight
 */
public inline fun
CharSequence.reduceRightIndexed(operation: (index: Int, Char, acc: Char) -> Char): Char {

    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}
```

```kotlin
/**
 * Accumulates value starting with the last character and applying [operation] from right to left
 * to each character with its index in the original char sequence and current accumulator value.
 * Returns `null` if the char sequence is empty.

 * @param [operation] function that takes the index of a character, the character itself and current accumulator value,
 * and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduceRightOrNull
 */
@SinceKotlin("1.4")
public inline fun
CharSequence.reduceRightIndexedOrNull(operation: (index: Int, Char, acc: Char) -> Char): Char? {

    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}
```

```kotlin
/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right
 * to each character and current accumulator value that starts with [initial] value.

 * @param [operation] function that takes a character and current accumulator value, and calculates
 * the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.runningFold
 */
@SinceKotlin("1.4")
public inline fun
CharSequence.runningFold(initial: R, operation: (acc: R, Char) -> R): List<R> {

    if (isEmpty()) return listOf(initial)
    val result = ArrayList<R>(length + 1).apply { add(initial) }
    var accumulator = initial
    for (element in this) {
        accumulator = operation(accumulator, element)
        result.add(accumulator)
    }
    return result
}
```

```kotlin
/**
 * Returns a list containing successive accumulation values generated by applying [operation]
 * from left to right to each character, its index in the original char sequence and current accumulator value
 * that starts with [initial] value.

 * @param [operation] function that takes the index of a character, current accumulator value
 * and the character itself, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.runningFoldIndexed
 */
@SinceKotlin("1.4")
public inline fun <R> CharSequence.runningFoldIndexed(initial: R, operation: (index: Int, acc: R, Char) -> R): List<R> {

    if (isEmpty()) return listOf(initial)
    val result = ArrayList<R>(length + 1).apply { add(initial) }
    var accumulator = initial
    for (index in indices) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}
```

```kotlin
/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right
 * to each character, its index in the original char sequence and current accumulator value.

 * @param [operation] function that takes the index of a character, current accumulator value
 * and the character itself, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.runningFold
 */
@SinceKotlin("1.4")
public inline fun <R> CharSequence.runningFoldIndexed(initial: R, operation: (index: Int, acc: R, Char) -> R): List<R> {

    if (isEmpty()) return listOf(initial)
    val result = ArrayList<R>(length + 1).apply { add(initial) }
    var accumulator = initial
    for (index in indices) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}
```

```kotlin
public inline fun
CharSequence.reduceRight

```
each character and current accumulator value that starts with the first character of this char sequence.

* Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.

* @param [operation] function that takes current accumulator value and a character, and calculates the next accumulator value.

samples.collections.Collections.Aggregates.runningReduce

```kotlin
@SinceKotlin("1.4")
public inline fun CharSequence.runningReduce(operation: (acc: Char, Char) -> Char): List<Char> {
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<Char>(length).apply {
        add(accumulator)
    }
    for (index in 1 until length) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}
```

```kotlin
/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right to each character, its index in the original char sequence and current accumulator value that starts with the first character of this char sequence.

 * Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.

 * @param [operation] function that takes the index of a character, current accumulator value and the character itself, and calculates the next accumulator value.

 * @sample samples.collections.Collections.Aggregates.scan
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public inline fun <R> CharSequence.scanIndexed(initial: R, operation: (index: Int, acc: R, Char) -> R): List<R> {
    return runningFoldIndexed(initial, operation)
}
```

```kotlin
@Deprecated("Use runningReduce instead.",
ReplaceWith("runningReduce(operation)"), level = DeprecationLevel.ERROR)
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public inline fun CharSequence.scanReduce(operation: (acc: Char, Char) -> Char): List<Char> {
    return runningReduce(operation)
}
```

```kotlin
@Deprecated("Use runningReduceIndexed instead.",
ReplaceWith("runningReduceIndexed(operation)"), level = DeprecationLevel.ERROR)
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public inline fun CharSequence.scanReduceIndexed(operation: (index: Int, acc: Char, Char) -> Char): List<Char> {
    return runningReduceIndexed(operation)
}
```

```kotlin
/**
 * Returns the sum of all values produced by [selector] function applied to each character in the char sequence.

 * @param [selector] function applied to each character in the char sequence.

 * @sample
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public inline fun CharSequence.sumBy(selector: (Char) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}
```

```kotlin
/**
 * Returns the sum of all values produced by [selector] function applied to each character in the char sequence.

 * @param [selector] function applied to each character in the char sequence.

 * @sample
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public inline fun CharSequence.sumByDouble(selector: (Char) -> Double): Double {
    var sum: Double = 0.0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}
```
all values produced by [selector] function applied to each character in the char sequence.\n
*\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType\n@kotlin.jvm.JvmName("sumOfDouble")\n@kotlin.internal.InlineOnly\n\npublic inline fun CharSequence.sumOf(selector: (Char) -> Double): Double {\n    var sum: Double = 0.toDouble()\n    for (element in this) {\n        sum += selector(element)\n    }\n    return sum\n}

/**\n * Returns the sum of all values produced by [selector] function applied to each character in the char sequence.\n
*\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType\n@kotlin.jvm.JvmName("sumOfInt")\n@kotlin.internal.InlineOnly\n\npublic inline fun CharSequence.sumOf(selector: (Char) -> Int): Int {\n    var sum: Int = 0.toInt()\n    for (element in this) {\n        sum += selector(element)\n    }\n    return sum\n}

/**\n * Returns the sum of all values produced by [selector] function applied to each character in the char sequence.\n
*\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType\n@kotlin.jvm.JvmName("sumOfLong")\n@kotlin.internal.InlineOnly\n\npublic inline fun CharSequence.sumOf(selector: (Char) -> Long): Long {\n    var sum: Long = 0.toLong()\n    for (element in this) {\n        sum += selector(element)\n    }\n    return sum\n}

/**\n * Splits this char sequence into a list of strings each not exceeding the given [size].\n * \n * The last string in the resulting list may have fewer characters than the given [size].\n * \n * @param size the number of elements to take in each string, must be positive and can be greater than the number of elements in this char sequence.\n * \n * @sample samples.text.Strings.chunked\n */\n@SinceKotlin("1.2")\n\npublic fun CharSequence.chunked(size: Int): List<String> {\n    return windowed(size, size, partialWindows = true)\n}

/**\n * Splits this char sequence into several char sequences each not exceeding the given [size] and applies the given [transform] function to an each.\n * \n * @param size the number of elements to take in each string, must be positive and can be greater than the number of elements in this char sequence.\n * \n * @sample samples.text.Strings.chunked\n */\n@SinceKotlin("1.2")\n\npublic fun <R> CharSequence.chunked(size: Int, transform: (CharSequence) -> R): List<R> {\n    return windowed(size, size, partialWindows = true, transform = transform)\n}

/**\n * Splits this char sequence into a sequence of strings each not exceeding the given [size] and applies the given [transform] function to an each.\n * \n * @param size the number of elements to take in each string, must be positive and can be greater than the number of elements in this char sequence.\n * \n * @sample samples.text.Strings.chunked\n */\n@SinceKotlin("1.2")\n\npublic fun String.chunked(size: Int, transform: (CharSequence) -> R): List<R> {\n    return windowed(size, size, partialWindows = true, transform = transform)\n}

*\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType\n@kotlin.jvm.JvmName("sumOfDouble")\n@kotlin.internal.InlineOnly\n\npublic inline fun CharSequence.sumOf(selector: (Char) -> Double): Double {\n    var sum: Double = 0.toDouble()\n    for (element in this) {\n        sum += selector(element)\n    }\n    return sum\n}

/**\n * Returns the sum of all values produced by [selector] function applied to each character in the char sequence.\n
*\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType\n@kotlin.jvm.JvmName("sumOfInt")\n@kotlin.internal.InlineOnly\n\npublic inline fun CharSequence.sumOf(selector: (Char) -> Int): Int {\n    var sum: Int = 0.toInt()\n    for (element in this) {\n        sum += selector(element)\n    }\n    return sum\n}

/**\n * Returns the sum of all values produced by [selector] function applied to each character in the char sequence.\n
*\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType\n@kotlin.jvm.JvmName("sumOfLong")\n@kotlin.internal.InlineOnly\n\npublic inline fun CharSequence.sumOf(selector: (Char) -> Long): Long {\n    var sum: Long = 0.toLong()\n    for (element in this) {\n        sum += selector(element)\n    }\n    return sum\n}

/**\n * Splits this char sequence into a list of strings each not exceeding the given [size].\n * \n * The last string in the resulting list may have fewer characters than the given [size].\n * \n * @param size the number of elements to take in each string, must be positive and can be greater than the number of elements in this char sequence.\n * \n * @sample samples.text.Strings.chunked\n */\n@SinceKotlin("1.2")\n\npublic fun CharSequence.chunked(size: Int): List<String> {\n    return windowed(size, size, partialWindows = true)\n}

/**\n * Splits this char sequence into several char sequences each not exceeding the given [size] and applies the given [transform] function to an each.\n * \n * @param size the number of elements to take in each string, must be positive and can be greater than the number of elements in this char sequence.\n * \n * @sample samples.text.Strings.chunked\n */\n@SinceKotlin("1.2")\n\npublic fun <R> CharSequence.chunked(size: Int, transform: (CharSequence) -> R): List<R> {\n    return windowed(size, size, partialWindows = true, transform = transform)\n}

/**\n * Splits this char sequence into a sequence of strings each not exceeding the given [size] and applies the given [transform] function to an each.\n * \n * @param size the number of elements to take in each string, must be positive and can be greater than the number of elements in this char sequence.\n * \n * @sample samples.text.Strings.chunked\n */\n@SinceKotlin("1.2")\n\npublic fun String.chunked(size: Int, transform: (CharSequence) -> R): List<R> {\n    return windowed(size, size, partialWindows = true, transform = transform)\n}
the given [size].

@param size the number of elements to take in each character sequence, must be positive and
can be greater than the number of elements in this character sequence.

@sample samples.text.Strings.chunkedTransformToSequence

public fun <R>
CharSequence.chunkedSequence(size: Int, transform: (CharSequence) -> R): Sequence<R> {
    return windowedSequence(size, size, partialWindows = true, transform = transform)
}

@sample samples.text.Strings.partition

public inline fun CharSequence.partition(predicate: (Char) -> Boolean): Pair<CharSequence, CharSequence> {
    val first = StringBuilder()
    val second = StringBuilder()
    for (element in this) {
        if (predicate(element)) {
            first.append(element)
        } else {
            second.append(element)
        }
    }
    return Pair(first, second)
}

public inline fun String.partition(predicate: (Char) -> Boolean): Pair<String, String> {
    val first = StringBuilder()
    val second = StringBuilder()
    for (element in this) {
        if (predicate(element)) {
            first.append(element)
        } else {
            second.append(element)
        }
    }
    return Pair(first.toString(), second.toString())
}

@SinceKotlin("1.2")
public fun CharSequence.windowed(size: Int, step: Int = 1, partialWindows: Boolean = false): List<String> {
    return windowed(size, step, partialWindows) { it.toString() }
}

@SinceKotlin("1.2")
public fun <R> CharSequence.windowed(size: Int, step: Int = 1, partialWindows: Boolean = false, transform: (CharSequence) -> R): List<R> {
    checkWindowSizeStep(size, step)
    val thisSize = this.length
    val resultCapacity = thisSize / step
    if (thisSize % step == 0) resultCapacity
    val result = ArrayList<R>(resultCapacity)
    var index = 0
    while (index < thisSize) {
        val end = index + size
        val coercedEnd = if (end < 0 || end > thisSize) { if (partialWindows) thisSize else break } else end
        result.add(transform(subSequence(index, coercedEnd)))
        index += step
    }
    return result
}

@SinceKotlin("1.2")
public fun
CharSequence.windowed(size: Int, step: Int = 1, partialWindows: Boolean = false): List<String> {
    return windowed(size, step, partialWindows) { it.toString() }
}

@SinceKotlin("1.2")
public fun
CharSequence.windowed(size: Int, step: Int = 1, partialWindows: Boolean = false, transform: (CharSequence) -> R): List<R> {
    checkWindowSizeStep(size, step)
    val thisSize = this.length
    val resultCapacity = thisSize / step
    if (thisSize % step == 0) resultCapacity
    val result = ArrayList<R>(resultCapacity)
    var index = 0
    while (index < thisSize) {
        val end = index + size
        val coercedEnd = if (end < 0 || end > thisSize) { if (partialWindows) thisSize else break } else end
        result.add(transform(subSequence(index, coercedEnd)))
        index += step
    }
    return result
}

Open Source Used In security_automation identity-guest-cs 1467
CharSequence.windowedSequence(size: Int, step: Int = 1, partialWindows: Boolean = false): Sequence<String> { 
    return windowedSequence(size, step, partialWindows) { 
        it.toString() 
    } 
}

/**
 * Returns a sequence of results of applying the given [transform] function to
each char sequence representing a view over the window of the
given [size].
 * sliding along this char sequence with the given [step].
 * Note that the char sequence passed to the [transform] function is ephemeral and is valid only inside that function.
 * You should not store it or allow it to escape in some way, unless you made a snapshot of it.
 * Several last char sequences may have fewer characters than the given [size].
 * Both [size] and [step] must be positive and can be greater than the number of elements in this char sequence.
 * @param size the number of elements to take in each window
 * @param step the number of elements to move the window forward by on each an step, by default 1
 * @param partialWindows controls whether or not to keep partial windows in the end if any.
 * by default `false` which means partial windows won't be preserved
 * The returned list has length of the shortest char sequence.
 * Returned is empty if this char sequence contains less than two characters.
 * The returned list is empty if this char sequence contains less than two characters.
 * The returned list is empty if this char sequence contains less than two characters.
 */
@SinceKotlin("1.2")
public fun <R> CharSequence.windowedSequence(size: Int, step: Int = 1, 
    partialWindows: Boolean = false, transform: (CharSequence) -> R): Sequence<R> {
    checkWindowSizeStep(size, step)
    val windows = (if (partialWindows) indices else 0 until length - size + 1) step step
    return windows.asSequence().map { index ->
        val end = index + size
        val coercedEnd = if (end < 0 || end > length) length else end
        transform(subSequence(index, coercedEnd))
    }
}

/**
 * Returns a list of pairs built from the characters of `this` and the [other] char sequences with the same index
 * The returned list is empty if this char sequence contains less than two characters.
 * The returned list is empty if this char sequence contains less than two characters.
 */
@SinceKotlin("1.2")
public fun infix fun CharSequence.zip(other: CharSequence): List<Pair<Char, Char>> {
    return zip(other) { c1, c2 -> c1 to c2 }
}

/**
 * Returns a list of values built from the characters of `this` and the [other] char sequences with the
 * the same index
 * using the provided [transform] function applied to each pair of characters.
 * The returned list is empty if this char sequence contains less than two characters.
 * The returned list is empty if this char sequence contains less than two characters.
 */
public inline fun <V> CharSequence.zip(other: CharSequence, transform: (a: Char, b: Char) -> V): List<V> {
    val length = minOf(this.length, other.length)
    val list = ArrayList<V>(length)
    for (i in 0 until length) {
        list.add(transform(this[i], other[i]))
    }
    return list
}

/**
 * Creates an [Iterable] instance that wraps the original char sequence returning its characters when being iterated.
 */
public fun CharSequence.asIterable(): Iterable<Char> {
    if (this is String && isEmpty()) return emptyList()
    return 
}

/**
 * Creates a [Sequence] instance that wraps the original char sequence returning its characters when being iterated.
 */
public fun CharSequence.asSequence(): Sequence<Char> {
    if (this is String && isEmpty()) return emptySequence()
    return Sequence { this.iterator() }
}

Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

Open Source Used In security_automation identity-guest-cs 1468

4

1468
Returns a copy of this string having its first letter titlecased using the rules of the default locale.

The title case of a character is usually the same as its upper case with several exceptions. The particular list of characters with the special title case form depends on the underlying platform.

Returns a subsequence of this char sequence having leading and trailing characters matching the [predicate] removed.

* `CharSequence.trim(predicate: (Char) -> Boolean): CharSequence`

    | `(Char) -> Boolean: CharSequence { (in this indices) } return subsequence(index, length))` |
    | `(Char) -> Boolean: CharSequence { (in this indices, reversed()) } return subsequence(0, index + 1)` |

* `CharSequence.trimStart(predicate: (Char) -> Boolean): CharSequence`

    | `(Char) -> Boolean: CharSequence { (in this indices) } return subsequence(index, length)` |
    | `(Char) -> Boolean: CharSequence { (in this indices, reversed()) } return subsequence(0, index + 1)` |

* `CharSequence.trimEnd(predicate: (Char) -> Boolean): CharSequence`

    | `(Char) -> Boolean: CharSequence { (in this indices) } return subsequence(index, length)` |
    | `(Char) -> Boolean: CharSequence { (in this indices, reversed()) } return subsequence(0, index + 1)` |

* `CharSequence.trim(vararg chars: Char): CharSequence`

    | `(Char) -> Boolean: CharSequence { (in this indices) } return subsequence(index, length)` |
    | `(Char) -> Boolean: CharSequence { (in this indices, reversed()) } return subsequence(0, index + 1)` |

* `CharSequence.trimStart(vararg chars: Char): CharSequence`

    | `(Char) -> Boolean: CharSequence { (in this indices) } return subsequence(index, length)` |
    | `(Char) -> Boolean: CharSequence { (in this indices, reversed()) } return subsequence(0, index + 1)` |

* `CharSequence.trimEnd(vararg chars: Char): CharSequence`

    | `(Char) -> Boolean: CharSequence { (in this indices) } return subsequence(index, length)` |
    | `(Char) -> Boolean: CharSequence { (in this indices, reversed()) } return subsequence(0, index + 1)` |
Returns a string having leading whitespace removed.

```kotlin
public inline fun String.trimStart(): String = (this as CharSequence).trimStart().toString()
```

Returns a string having trailing whitespace removed.

```kotlin
public inline fun String.trimEnd(): String = (this as CharSequence).trimEnd().toString()
```

Returns a char sequence with content of this char sequence padded at the beginning.

```kotlin
public fun CharSequence.padStart(length: Int, padChar: Char = ' '): CharSequence {
    // Implementation...
}
```

Returns a string of length at least [length] consisting of `this` char sequence padded with [padChar] as many times as necessary to reach that length.

```kotlin
public inline fun CharSequence?.isNullOrEmpty(): Boolean {
    // Implementation...
}
```

Returns `true` if this nullable char sequence is either `null` or empty.

```kotlin
public inline fun CharSequence.isEmpty(): Boolean = length == 0
```

Returns `true` if this char sequence is not empty.

```kotlin
public inline fun CharSequence.isNotBlank(): Boolean = !isBlank()
```

Returns `true` if this nullable char sequence is either `null` or empty or consists solely of whitespace characters.

```kotlin
public inline fun CharSequence.isNotBlank(): Boolean = !isBlank()
```

Returns `true` if this nullable char sequence is either `null` or empty and contains no characters.

```kotlin
public inline fun CharSequence.isEmptyOrNullOrEmpty(): Boolean {
    // Implementation...
}
```

Returns `true` if this char sequence is empty (contains no characters).

```kotlin
public inline fun CharSequence.isNotBlankOrNullOrEmpty(): Boolean {
    // Implementation...
}
```

Returns `true` if this char sequence is not empty and contains no characters.
```kotlin
CharSequence?.isNullOrBlank(): Boolean { un contract { un returns(false) implies (this@isNullOrBlank !=
null) un } un return this == null || this.isBlank() un
}

public operator fun CharSequence.iterator(): CharIterator = object : CharIterator() { un private var index = 0 un
public override fun nextChar(): Char = get(index++) un
public override fun hasNext(): Boolean = index <
length un
}

/**
 * Iterator for characters of the given char sequence.
 */
public operator fun CharSequence.iterator(): CharIterator = object : CharIterator() {

private var index = 0

public override fun nextChar(): Char = get(index++)

public override fun hasNext(): Boolean = index <
length

/**
 * Returns the range of valid character indices for this char sequence.
 */
public val CharSequence.indices: IntRange
    get() = 0..length - 1

/**
 * Returns the index of the last character in the char sequence or -1 if it is empty.
 */
public val CharSequence.lastIndexOf: Int
    get() = this.length - 1

/**
 * Returns `true` if this CharSequence has Unicode surrogate pair at the specified [index].
 */
public fun CharSequence.hasSurrogatePairAt(index: Int): Boolean {

    return index in 0..length - 2
            && this[index].isHighSurrogate()
            && this[index + 1].isLowSurrogate()
}

/**
 * Returns a substring specified by the given [range] of indices.
 */
public fun String.substring(range: IntRange): String = substring(range.start, range.endInclusive + 1)

/**
 * Returns a string if it's not empty or the result of calling [defaultValue] function if the char sequence is empty.
 */
@sample samples.text.Strings.stringIfEmpty

@SinceKotlin("1.3")
@sample @kotlin.internal.InlineOnly
@public inline fun String.orEmpty(): String = this ?: ""

/**
 * Returns this char sequence if it's not empty and doesn't consist solely of whitespace characters,
 or the result of calling [defaultValue] function otherwise.
 */
@sample samples.text.Strings.stringIfBlank

@SinceKotlin("1.3")
@sample @kotlin.internal.InlineOnly
@public inline fun <C, R> C.ifEmpty(defaultValue: () -> R): R where C : CharSequence, C : R =
    if (isEmpty()) defaultValue() else this

/**
 * Returns a substring before the first occurrence of [delimiter].
 * If the string does not contain the delimiter, returns [missingDelimiterValue] which defaults to the original string.
 */
public inline fun <C, R> C.substringBefore(delimiter: Char, missingDelimiterValue: String = this): String {
    val index = indexOf(delimiter)
    return if (index == -1) missingDelimiterValue else substring(0, index)
}

/**
 * Returns a substring before the first occurrence of [delimiter].
 * If the string does not contain the delimiter, returns [missingDelimiterValue] which defaults to the original string.
 */
public inline fun <C, R> C.substringBefore(delimiter: String, missingDelimiterValue: String = this): String {
    val index = indexOf(delimiter)
    return if (index == -1) missingDelimiterValue else substring(0, index)
}

/**
 * Returns a substring after the first occurrence of [delimiter].
 * If the string does not contain the delimiter, returns [missingDelimiterValue] which defaults to the original string.
 */
public inline fun <C, R> C.substringAfter(delimiter: Char, missingDelimiterValue: String = this): String {
    val index = indexOf(delimiter)
    return if (index == -1) missingDelimiterValue else substring(index + 1, length)
}

/**
 * Returns a substring after the first occurrence of [delimiter].
 * If the string does not contain the delimiter, returns [missingDelimiterValue] which defaults to the original string.
 */
public inline fun <C, R> C.substringAfter(delimiter: String, missingDelimiterValue: String = this): String {
    val index = indexOf(delimiter)
    return if (index == -1) missingDelimiterValue else substring(index + 1, length)
}
```

Open Source Used In security_automation identity-guest-cs 1471
String {\n    val index = indexOf(delimiter)\n    return if (index == -1) missingDelimiterValue else substring(index + delimiter.length, length)\n}\n
/**
 * Returns a substring before the last occurrence of [delimiter].
 * @param delimiter The delimiter character.
 * @param missingDelimiterValue The value to return if the string does not contain the delimiter.
 */
public fun String.substringBeforeLast(delimiter: Char, missingDelimiterValue: String = this): String {
    val index = lastIndexOf(delimiter)
    return if (index == -1) missingDelimiterValue else substring(0, index)
}

/**
 * Returns a substring before the last occurrence of [delimiter].
 * @param delimiter The delimiter string.
 * @param missingDelimiterValue The value to return if the string does not contain the delimiter.
 */
public fun String.substringBeforeLast(delimiter: String, missingDelimiterValue: String = this): String {
    val index = lastIndexOf(delimiter)
    return if (index == -1) missingDelimiterValue else substring(0, index)
}

/**
 * Returns a substring after the last occurrence of [delimiter].
 * @param delimiter The delimiter character.
 * @param missingDelimiterValue The value to return if the string does not contain the delimiter.
 */
public fun String.substringAfterLast(delimiter: Char, missingDelimiterValue: String = this): String {
    val index = lastIndexOf(delimiter)
    return if (index == -1) missingDelimiterValue else substring(index + 1, length)
}

/**
 * Returns a substring after the last occurrence of [delimiter].
 * @param delimiter The delimiter string.
 * @param missingDelimiterValue The value to return if the string does not contain the delimiter.
 */
public fun String.substringAfterLast(delimiter: String, missingDelimiterValue: String = this): String {
    val index = lastIndexOf(delimiter)
    return if (index == -1) missingDelimiterValue else substring(index + delimiter.length, length)
}

/**
 * Replaces the part of the string at the given range with the [replacement] string.
 * @param range The range of characters to be replaced.
 * @param replacement The replacement string.
 */
public fun String.replaceRange(range: IntRange, replacement: String): String =
    (this as CharSequence).replaceRange(range, replacement).toString()

/**
 * Removes the part of string at a given range.
 * @param range The range of characters to be removed.
 * @param endInclusive Whether the end index of the range is included.
 */
public fun String.removeRange(startIndex: Int, endIndex: Int): String =
    (this as CharSequence).removeRange(startIndex, endIndex).toString()
given [range] is removed.\n *\n* \nThe end index of the [range] is included in the removed part.\n
```kotlin
//\n*\n* public fun CharSequence.removeRange(range: IntRange): CharSequence = removeRange(range.start, range.endInclusive + 1)\n```

*\n* Removes the part of a string at the given [range].\n*\n* The end index of the [range] is included in the removed part.\n
```kotlin
//\n*\n* This as CharSequence).removeRange(range).toString()\n```

*\n* If this char sequence starts with the given [prefix], returns a new char sequence\n*\n* with the prefix removed. Otherwise, returns a new char sequence with the same characters.\n
```kotlin
//\n*\n* public fun CharSequence.removePrefix(prefix: CharSequence): CharSequence {\n```

```java
//\n*\n* return subSequence(0, length - prefix.length)\n```

*\n* Returns a copy of this string\n*\n* with the prefix removed. Otherwise, returns this string.\n
```kotlin
//\n*\n* public fun String.removePrefix(prefix: CharSequence): String {\n```

```kotlin
//\n*\n* substring(0, length - prefix.length)\n```

*\n* Returns this\n*\n* with the suffix removed. Otherwise, returns a new char sequence\n*\n* with the same characters.\n
```kotlin
//\n*\n* public fun CharSequence.removeSurrounding(prefix: CharSequence, suffix: CharSequence): CharSequence {\n```

```kotlin
//\n*\n* return subSequence(prefix.length, length - suffix.length)\n```

*\n* Removes from a string both the given [prefix] and [suffix] if and only if\n*\n* it starts with the [prefix] and ends with the [suffix].\n
```kotlin
//\n*\n* return subSequence(0, length)\n```

*\n* If this char sequence starts with the given [prefix] and ends with the given [suffix],\n*\n* returns a new char sequence having both the given [prefix] and [suffix] removed.\n*\n* Otherwise returns a new char sequence with the same characters.\n
```kotlin
//\n*\n* public fun CharSequence.removePrefix(suffix: CharSequence): CharSequence {
```

```kotlin
//\n*\n* length - suffix.length\n```

*\n* Returns this\n*\n* with the suffix removed. Otherwise, returns a new char sequence\n*\n* with the same characters.\n
```kotlin
//\n*\n* public fun CharSequence.removeSuffix(suffix: CharSequence): CharSequence {
```

```kotlin
//\n*\n* return subSequence(0, length)\n```

*\n* If this char sequence ends with the given [suffix], returns a new char sequence\n*\n* with the suffix removed. Otherwise, returns a new char sequence\n*\n* with the same characters.\n
```kotlin
//\n*\n* public fun String.removeSuffix(suffix: CharSequence): String {
```

```kotlin
//\n*\n* substring(0, length - suffix.length)\n```

*\n* Returns this\n*\n* with the suffix removed. Otherwise, returns a new char sequence\n*\n* with the same characters.\n
```kotlin
//\n*\n* public fun String.removeSurrounding(delimiter: CharSequence): String = removeSurrounding(delimiter, delimiter)\n```

```kotlin
//\n*\n* return subSequence(0, length)\n```

*\n* Remove the delimiter from both the start and end.\n*\n* Otherwise returns a new char sequence with the same characters.\n
```kotlin
//\n*\n* return subSequence(0, length)\n```

*\n* If this char sequence starts with the given [delimiter] and ends with the given [delimiter],\n*\n* returns a new char sequence having this [delimiter] removed from both the start and end.\n*\n* Otherwise returns a new char sequence with the same characters.\n
```kotlin
//\n*\n* return subSequence(0, length)\n```

*\n* If this char sequence starts with the given [delimiter] and ends with the given [delimiter],\n*\n* returns a new char sequence having this [delimiter] removed from both the start and end.\n*\n* Otherwise returns a new char sequence with the same characters.\n
```kotlin
//\n*\n* return subSequence(0, length)\n```

*\n* If this string does not contain the delimiter, returns\n*\n* [missingDelimiterValue] which defaults to the original string.\n
```kotlin
//\n*\n* return subSequence(0, length)\n```

*\n* If the string does not contain the delimiter, returns\n*\n* [missingDelimiterValue] which defaults to the original string.\n
```kotlin
//\n*\n* return subSequence(0, length)\n```

*\n* If the string does not contain the delimiter, returns\n*\n* [missingDelimiterValue] which defaults to the original string.\n
```kotlin
//\n*\n* return subSequence(0, length)\n```

*\n* If the string does not contain the delimiter, returns\n*\n* [missingDelimiterValue] which defaults to the original string.\n
```kotlin
//\n*\n* return subSequence(0, length)\n```

*\n* If the string does not contain the delimiter, returns\n*\n* [missingDelimiterValue] which defaults to the original string.\n
```kotlin
//\n*\n* return subSequence(0, length)\n```

*\n* If the string does not contain the delimiter, returns\n*\n* [missingDelimiterValue] which defaults to the original string.
replacement: String, missingDelimiterValue: String = this): String {
    val index = indexOf(delimiter)
    return if (index == -1) missingDelimiterValue else replaceRange(index + delimiter.length, length, replacement)
}

/**
 * Replace part of string after the last occurrence of given delimiter with the [replacement] string.
 * If the string does not contain the delimiter, returns [missingDelimiterValue] which defaults to the original string.
 */
public fun String.replaceAfterLast(delimiter: String, replacement: String, missingDelimiterValue: String = this): String {
    val index = lastIndexOf(delimiter)
    return if (index == -1) missingDelimiterValue else replaceRange(index + delimiter.length, length, replacement)
}

/**
 * Replace part of string after the last occurrence of given delimiter with the [replacement] string.
 * If the string does not contain the delimiter, returns [missingDelimiterValue] which defaults to the original string.
 */
public fun String.replaceAfterLast(delimiter: Char, replacement: String, missingDelimiterValue: String = this): String {
    val index = lastIndexOf(delimiter)
    return if (index == -1) missingDelimiterValue else replaceRange(index + 1, length, replacement)
}

/**
 * Replace part of string before the last occurrence of given delimiter with the [replacement] string.
 * If the string does not contain the delimiter, returns [missingDelimiterValue] which defaults to the original string.
 */
public fun String.replaceBeforeLast(delimiter: Char, replacement: String, missingDelimiterValue: String = this): String {
    val index = lastIndexOf(delimiter)
    return if (index == -1) missingDelimiterValue else replaceRange(0, index, replacement)
}

/**
 * Replace part of string before the last occurrence of given delimiter with the [replacement] string.
 * If the string does not contain the delimiter, returns [missingDelimiterValue] which defaults to the original string.
 */
public fun String.replaceBeforeLast(delimiter: String, replacement: String, missingDelimiterValue: String = this): String {
    val index = lastIndexOf(delimiter)
    return if (index == -1) missingDelimiterValue else replaceRange(0, index, replacement)
}

/**
 * Returns a new string obtained by replacing each substring of this char sequence that matches the given regular expression
 * with the given [replacement].
 * The [replacement] can consist of any combination of literal text and $-substitutions. To treat the
 */
@kotlin.internal.InlineOnly
public inline fun CharSequence.replace(regex: Regex, replacement: String): String = regex.replace(this, replacement)

/**
 * Returns a new string obtained by replacing each substring of this char sequence that matches the given regular expression
 * with the result of the given function [transform] that takes [MatchResult] and returns a string to be used as
 * a replacement for that match.
 */
@kotlin.internal.InlineOnly
public inline fun CharSequence.replace(regex: Regex, noinline transform: (MatchResult) -> CharSequence): String =
    regex.replace(this, transform)

/**
 * Replaces the first occurrence of the given regular expression [regex] in this char sequence with specified [replacement] expression.
 * @param replacement A replacement expression that can include substitutions. See [Regex.replaceFirst] for details.
 * @sample samples.text.Strings.replaceFirstChar
 */
@sample samples.text.Strings.replaceFirstChar
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolutionByLambdaReturnType
@JvmName("replaceFirstCharWithChar")
@kotlin.internal.InlineOnly
public inline fun String.replaceFirstChar(transform: (Char) -> Char): String {
    return if (isNotEmpty()) transform(this[0]) + substring(1) else this
}

/**
 * Replaces the first occurrence of the given regular expression [regex] in this char sequence with specified [replacement] expression.
 * @param replacement A replacement expression that can include substitutions. See [Regex.replaceFirst] for details.
 * @sample samples.text.Strings.replaceFirstChar
 */
@sample samples.text.Strings.replaceFirstChar
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolutionByLambdaReturnType
@JvmName("replaceFirstCharWithCharSequence")
@kotlin.internal.InlineOnly
public inline fun String.replaceFirstChar(transform: (Char) -> CharSequence): String {
    return if (isNotEmpty()) transform(this[0]) + substring(1) else this
}
return if (isNotEmpty()) transform(this[0]).toString() + substring(1) else this

/**
 * Returns `true` if this char sequence matches the given regular expression.
 */
public fun matches(regex: Regex): Boolean = regex.matches(this)

/**
 * Returns `true` if this char sequence starts with the specified prefix.
 */
public fun startsWith(prefix: CharSequence, ignoreCase: Boolean = false): Boolean {
    if (!ignoreCase && this is String && prefix is String)
        return this.startsWith(prefix)
    else
        return regionMatchesImpl(0, prefix, 0, prefix.length, ignoreCase)
}

/**
 * Returns `true` if a substring of this char sequence starting at the specified offset [startIndex] starts with the specified prefix.
 */
public fun startsWith(prefix: CharSequence, startIndex: Int, ignoreCase: Boolean = false): Boolean {
    if (!ignoreCase && this is String && prefix is String)
        return this.startsWith(prefix, startIndex)
    else
        return regionMatchesImpl(startIndex, prefix, 0, prefix.length, ignoreCase)
}

/**
 * Returns `true` if this char sequence starts with the specified prefix.
 */
public fun startsWith(char: Char, ignoreCase: Boolean = false): Boolean =
    this.length > 0 && this[0].equals(char, ignoreCase)

/**
 * Returns `true` if this char sequence ends with the specified suffix.
 */
public fun endsWith(suffix: CharSequence, ignoreCase: Boolean = false): Boolean {
    if (!ignoreCase && this is String && suffix is String)
        return this.endsWith(suffix)
    else
        return regionMatchesImpl(length - suffix.length, suffix, 0, suffix.length, ignoreCase)
}

/**
 * Returns the longest string `prefix` such that this char sequence and [other] char sequence both start with this prefix,
 * taking care not to split surrogate pairs.
 * If this and [other] have no common prefix, returns the empty string.
 * @param ignoreCase `true` to ignore character case when matching a character.
 * @return String
 */
public fun commonPrefixWith(other: CharSequence, ignoreCase: Boolean = false): String {
    val shortestLength = minOf(this.length, other.length)
    var i = 0
    while (i < shortestLength && this[i].equals(other[i], ignoreCase))
        i++
    if (this.hasSurrogatePairAt(i - 1) || other.hasSurrogatePairAt(i - 1))
        i--
    return substrSequence(0, i).toString()
}

/**
 * Returns the longest string `suffix` such that this char sequence and [other] char sequence both end with this suffix,
 * taking care not to split surrogate pairs.
 * If this and [other] have no common suffix, returns the empty string.
 * @param ignoreCase `true` to ignore character case when matching a character.
 * @return String
 */
public fun commonSuffixWith(other: CharSequence, ignoreCase: Boolean = false): String {
    val thisLength = this.length
    val otherLength = other.length
    var i = 0
    while (i < shortestLength && this[i].equals(other[i], ignoreCase))
        i++
    if (this.hasSurrogatePairAt(i - 1) || other.hasSurrogatePairAt(i - 1))
        i--
    return substrSequence(thisLength - i, thisLength).toString()
}

/**
 * Finds the index of the first occurrence of any of the specified [chars] in this char sequence.
 * Optionally ignoring the specified [startIndex] and optionally ignoring the specified character.
 * @param ignoreCase `true` to ignore character case when matching a character.
 * @param startIndex the start index
 * @return Int
 */
public fun indexOfAny(chars: CharArray, startIndex: Int = 0, ignoreCase: Boolean = false): Int {
    for (index in startIndex.coerceAtLeast(0) until length)
        if (chars.any { it.equals(this[index], ignoreCase) })
            return index
}

// indexOfAny()

/**
 * Finds the index of the first occurrence of any of the specified [chars] in this char sequence,
 * optionally ignoring the specified [startIndex] and optionally ignoring the specified character.
 * @param ignoreCase `true` to ignore character case when matching a character.
 * @param startIndex the start index
 * @return Int
 */
public fun nativeIndexOf(char: Char, startIndex: Int = 0, ignoreCase: Boolean = false): Int {
    if (!ignoreCase && this.length < 0)
        return false
    for (index in 0 until length)
        if (!this[index].equals(char, ignoreCase))
            return false
    return true
}

// indexOfAny()

findAnyOf(strings, startIndex, ignoreCase, last = false)?.first ?: -1

/**
 * Finds the index of the last occurrence of any of the specified [strings] in this char sequence, starting from the specified [startIndex] and optionally ignoring the case.
 * @param strings A list of strings to search for.
 * @param startIndex The index to start the search from.
 * @param ignoreCase `true` to ignore character case when matching a string.
 * @param last `true` to return the index of the last occurrence of any of the specified strings.
 * @return An index of the last occurrence of a string from [strings] or -1 if none is found.
 */
public fun CharSequence.lastIndexOfAny(strings: Collection<String>, startIndex: Int = lastIndex, ignoreCase: Boolean = false): Int =
    findAnyOf(strings, startIndex, ignoreCase, last = true)?.first ?: -1

/**
 * Returns the index within this string of the first occurrence of the specified character, starting from the specified [startIndex].
 * @param ignoreCase `true` to ignore character case when matching a character.
 * @return An index of the first occurrence of a character or -1 if none is found.
 */
public fun CharSequence.indexOf(char: Char, startIndex: Int = 0, ignoreCase: Boolean = false): Int {
    return if (ignoreCase)
        indexOfAny(charArrayOf(char), startIndex, ignoreCase)
    else
        nativeIndexOf(char, startIndex)
}

/**
 * Returns the index within this char sequence of the first occurrence of the specified string, starting from the specified [startIndex].
 * @param ignoreCase `true` to ignore character case when matching a string.
 * @return An index of the first occurrence of a string or -1 if none is found.
 */
public fun CharSequence.indexOf(string: String, startIndex: Int = 0, ignoreCase: Boolean = false): Int {
    return if (ignoreCase)
        indexOf(string, startIndex, length, ignoreCase)
    else
        nativeIndexOf(string, startIndex)
}

/**
 * Returns the index within this char sequence of the last occurrence of the specified character, starting from the specified [startIndex].
 * @param ignoreCase `true` to ignore character case when matching a character.
 * @return An index of the last occurrence of a character or -1 if none is found.
 */
public fun CharSequence.lastIndexOf(char: Char, startIndex: Int = lastIndex, ignoreCase: Boolean = false): Int {
    return if (ignoreCase)
        lastIndexOfAny(charArrayOf(char), startIndex, ignoreCase)
    else
        nativeLastIndexOf(char, startIndex)
}

/**
 * Returns the index within this char sequence of the last occurrence of the specified string, starting from the specified [startIndex].
 * @param ignoreCase `true` to ignore character case when matching a string.
 * @return An index of the last occurrence of a string or -1 if none is found.
 */
public fun CharSequence.lastIndexOf(string: String, startIndex: Int = lastIndex, ignoreCase: Boolean = false): Int {
    return if (ignoreCase)
        indexOf(string, startIndex, 0, ignoreCase, last = true)
    else
        nativeLastIndexOf(string, startIndex)
}

/**
 * Returns `true` if this char sequence contains the specified [other] sequence of characters as a substring.
 * @param ignoreCase `true` to ignore character case when comparing strings.
 * @return `true` if this char sequence contains the specified string.
 */
@Suppress("INAPPLICABLE_OPERATOR_MODIFIER")
public operator fun CharSequence.contains(other: CharSequence, ignoreCase: Boolean = false): Boolean =
    if (other is String)
        indexOf(other, ignoreCase = ignoreCase) >= 0
    else
        indexOf(other, 0, length, ignoreCase) >= 0

/**
 * Returns `true` if this char sequence contains at least one match of the specified regular expression.
 * @param regex The regular expression to search for.
 * @return `true` if this char sequence contains at least one match of the specified regular expression.
 */
@kotlin.internal.InlineOnly
public inline operator fun CharSequence.contains(regex: Regex): Boolean =
    regex.containsMatchIn(this)

open fun operator fun CharSequence.contains(other: CharSequence, ignoreCase: Boolean = false): Boolean {
    return other in this
}

open fun operator fun CharSequence.contains(regex: Regex): Boolean {
    return regex.containsMatchIn(this)
}

open fun operator fun CharSequence.contains(other: CharSequence, ignoreCase: Boolean = false): Boolean {
    return other in this
}

open fun operator fun CharSequence.contains(regex: Regex): Boolean {
    return regex.containsMatchIn(this)
}

open fun operator fun CharSequence.contains(other: CharSequence, ignoreCase: Boolean = false): Boolean {
    return other in this
}

open fun operator fun CharSequence.contains(regex: Regex): Boolean {
    return regex.containsMatchIn(this)
}

open fun operator fun CharSequence.contains(other: CharSequence, ignoreCase: Boolean = false): Boolean {
    return other in this
}

open fun operator fun CharSequence.contains(regex: Regex): Boolean {
    return regex.containsMatchIn(this)
}

open fun operator fun CharSequence.contains(other: CharSequence, ignoreCase: Boolean = false): Boolean {
    return other in this
}

open fun operator fun CharSequence.contains(regex: Regex): Boolean {
    return regex.containsMatchIn(this)
}

open fun operator fun CharSequence.contains(other: CharSequence, ignoreCase: Boolean = false): Boolean {
    return other in this
}

open fun operator fun CharSequence.contains(regex: Regex): Boolean {
    return regex.containsMatchIn(this)
}

override fun operator set():
CharSequence.split(vararg delimiters: String, ignoreCase: Boolean = false, limit: Int = 0): List<String> { 
    if (delimiters.size == 1) { 
        val delimiter = delimiters[0]
        return split(delimiter, ignoreCase, limit)
    }

    return rangesDelimitedBy(delimiters, ignoreCase = ignoreCase, limit = limit).asIterable().map { substring(it) }
}

/**
 * Splits this char sequence to a list of strings around occurrences of the specified [delimiters].
 */
public fun CharSequence.split(delimiter: Char, ignoreCase: Boolean = false, limit: Int = 0): List<String> {
    if (delimiter.size == 1) {
        return split(delimiter.toString(), ignoreCase, limit)
    }

    return rangesDelimitedBy(delimiter, ignoreCase = ignoreCase, limit = limit).asIterable().map { substring(it) }
}

/**
 * Splits this char sequence to a sequence of strings around occurrences of the specified [delimiters].
 */
@kotlin.internal.InlineOnly
public inline fun CharSequence.split(regex: Regex, limit: Int = 0): List<String> = regex.split(this, limit)

/**
 * Splits this char sequence to a list of lines delimited by any of the following character sequences:
 * CRLF, LF or CR.
 */
public fun CharSequence.lineSequence(): Sequence<String> = splitToSequence("\r\n", "\n", "\r")

/**
 * Splits this char sequence to a list of lines delimited by any of the following character sequences:
 * CRLF, LF or CR.
 */
public fun CharSequence.lines(): List<String> = lineSequence().toList()
CharArray, offset: Int, length: Int): String {
    if (offset < 0 || length < 0 || chars.size - offset < length)
        throw IndexOutOfBoundsException("size: $chars.size; offset: $offset; length: $length")
    var result = 
    for (index in offset until offset + length) {
        result += chars[index]
    }
    return result
}

/**
 * Concatenates characters in this [CharArray] into a String.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public actual fun CharArray.concatToString(): String {
    var result = ""
    for (char in this) {
        result += char
    }
    return result
}

/**
 * Concatenates characters in this [CharArray] or its subrange into a String.
 *
 * @param startIndex the beginning (inclusive) of the subrange of characters, 0 by default.
 * @param endIndex the end (exclusive) of the subrange of characters, size of this array by default.
 * @throws IndexOutOfBoundsException if [startIndex] is less than zero or [endIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [startIndex] is greater than [endIndex].
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun CharArray.concatToString(startIndex: Int = 0,
    endIndex: Int = this.size): String {
    AbstractList.checkBoundsIndexes(startIndex, endIndex, this.size)
    var result = ""
    for (index in startIndex until endIndex) {
        result += this[index]
    }
    return result
}

/**
 * Returns a [CharArray] containing characters of this string.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public actual fun String.toCharArray(): CharArray {
    return CharArray(length) { get(it) }
}

/**
 * Returns a [CharArray] containing characters of this string or its substring.
 *
 * @param startIndex the beginning (inclusive) of the substring, 0 by default.
 * @param endIndex the end (exclusive) of the substring, length of this string by default.
 * @throws IndexOutOfBoundsException if [startIndex] is less than zero or [endIndex] is greater than the length of this string.
 * @throws IllegalArgumentException if [startIndex] is greater than [endIndex].
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun String.toCharArray(startIndex: Int = 0,
    endIndex: Int = this.length): CharArray {
    AbstractList.checkBoundsIndexes(startIndex, endIndex, length)
    return CharArray(endIndex - startIndex) { get(startIndex + it) }
}

/**
 * Decodes a string from the bytes in UTF-8 encoding in this array.
 *
 * Malformed byte sequences are replaced by the replacement char `\uFFFD`.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public actual fun ByteArray.decodeToString(): String {
    return decodeUtf8(this, 0, size, false)
}

/**
 * Decodes a string from the bytes in UTF-8 encoding in this array or its subrange.
 *
 * @param startIndex the beginning (inclusive) of the substring to decode, 0 by default.
 * @param endIndex the end (exclusive) of the substring to decode, size of this array by default.
 * @param throwOnInvalidSequence specifies whether to throw an exception on malformed byte sequence or replace.
 * @throws IndexOutOfBoundsException if [startIndex] is less than zero or [endIndex] is greater than the size of this array.
 * @throws CharacterCodingException if the byte array contains malformed UTF-8 byte sequence and [throwOnInvalidSequence] is true.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun ByteArray.decodeToString(
    startIndex: Int = 0,
    endIndex: Int = this.size,
    throwOnInvalidSequence: Boolean = false
): String {
    AbstractList.checkBoundsIndexes(startIndex, endIndex, size)
    return decodeUtf8(this, startIndex, endIndex, throwOnInvalidSequence)
}

/**
 * Encodes this string to an array of bytes in UTF-8 encoding.
 * @param startIndex the beginning (inclusive) of the substring to encode, 0 by default.
 * @param endIndex the end (exclusive) of the substring to encode, length of this string by default.
 * @param throwOnInvalidSequence specifies whether to throw an exception on malformed byte sequence or replace.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public actual fun String.encodeToByteArray(): ByteArray {
    return encodeUtf8(this, 0, length, false)
}

/**
 * Encodes this string or its substring to an array of bytes in UTF-8 encoding.
 * @param throwOnInvalidSequence specifies whether to throw an exception on malformed char sequence or replace.
 * @throws IndexOutOfBoundsException if [startIndex] is less than zero
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public actual fun String.encodeToByteArray(
    startIndex: Int = 0,
    endIndex: Int = this.length,
    throwOnInvalidSequence: Boolean = false
): ByteArray {
    AbstractList.checkBoundsIndexes(startIndex, endIndex, this.size)
    return encodeUtf8(this, startIndex, endIndex, throwOnInvalidSequence)
}

/*An@SinceKotlin("1.4")@WasExperimental(ExperimentalStdlibApi::class)@public actual fun CharArray.toString(): String {
    var result = ""
    for (char in this) {
        result += char
    }
    return result
}*/

/*An@SinceKotlin("1.4")@WasExperimental(ExperimentalStdlibApi::class)@public actual fun CharArray.concatToString(): String {
    var result = ""
    for (index in offset until offset + length) {
        result += String(chars[index])
    }
    return result
}*/

/*An@SinceKotlin("1.4")@WasExperimental(ExperimentalStdlibApi::class)@public actual fun String.toCharArray(): CharArray {
    return CharArray(length) { get(it) }
}*/

/*An@SinceKotlin("1.4")@WasExperimental(ExperimentalStdlibApi::class)@public actual fun String.concatToString(): CharArray {
    var result = ""
    for (char in this) {
        result += char
    }
    return result
}*/

/*An@SinceKotlin("1.4")@WasExperimental(ExperimentalStdlibApi::class)@public actual fun String.concatToString(startIndex: Int = 0,
    endIndex: Int = this.size): CharArray {
    AbstractList.checkBoundsIndexes(startIndex, endIndex, this.size)
    return CharArray(endIndex - startIndex) { get(startIndex + it) }
}*/

/*An@SinceKotlin("1.4")@WasExperimental(ExperimentalStdlibApi::class)@public actual fun String.concatToString(startIndex: Int = 0,
    endIndex: Int = this.size, throwOnInvalidSequence: Boolean = false
): String {
    AbstractList.checkBoundsIndexes(startIndex, endIndex, this.size)
    return decodeUtf8(this, startIndex, endIndex, throwOnInvalidSequence)
}*/

/*An@SinceKotlin("1.4")@WasExperimental(ExperimentalStdlibApi::class)@public actual fun String.encodeToByteArray(): ByteArray {
    return encodeUtf8(this, 0, length, false)
}*/

/*An@SinceKotlin("1.4")@WasExperimental(ExperimentalStdlibApi::class)@public actual fun String.encodeToByteArray(startIndex: Int = 0,
    endIndex: Int = this.size, throwOnInvalidSequence: Boolean = false
): ByteArray {
    AbstractList.checkBoundsIndexes(startIndex, endIndex, this.size)
    return encodeUtf8(this, startIndex, endIndex, throwOnInvalidSequence)
}*/

Open Source Used In security_automation identity-guest-cs 1480
or [endIndex] is greater than the length of this string.

* @throws IllegalArgumentException if [startIndex] is greater than [endIndex].
* @throws CharacterCodingException if this string contains malformed char sequence and [throwOnInvalidSequence] is true.

* @SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
*public actual fun String.encodeToByteArray(
    startIndex: Int = 0,
    endIndex: Int = this.length,
    throwOnInvalidSequence: Boolean = false
): ByteArray {

AbstractList.checkBoundsIndexes(startIndex, endIndex, length)
    return encodeUtf8(this, startIndex, endIndex, throwOnInvalidSequence)
}

/**
 * Returns a copy of this string converted to upper case using the rules of the default locale.
 * @n * @n
 */
@SinceKotlin("1.4")
@ExperimentalStdlibApi
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun String.uppercase(): String = asDynamic().toUpperCase()

/**
 * Returns a copy of this string converted to upper case using Unicode mapping rules of the invariant locale.
 * This function supports one-to-many and many-to-one character mapping.
 * thus the length of the returned string can be different from the length of the original string.
 */
@SinceKotlin("1.4")
@ExperimentalStdlibApi
public actual inline fun String.uppercase(): String = asDynamic().toUpperCase()

/**
 * Returns a copy of this string converted to lower case using the rules of the default locale.
 */
@kotlin.internal.InlineOnly
public actual inline fun String.toLowerCase(): String = asDynamic().toLowerCase()

/**
 * Returns a copy of this string converted to lower case using Unicode mapping rules of the invariant locale.
 * This function supports one-to-many and many-to-one character mapping.
 * thus the length of the returned string can be different from the length of the original string.
 */
@SinceKotlin("1.4")
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
public actual inline fun String.lowercase(): String = asDynamic().toLowerCase()
Apache 2.0 license that can be found in the license/LICENSE.txt file.

package kotlin
    @SinceKotlin("1.3")
    @ExperimentalUnsignedTypes
    public inline class UByteArray
            @Suppress("NON_PUBLIC_PRIMARY_CONSTRUCTOR_OF_INLINE_CLASS")
    @PublishedApi
    internal constructor(@PublishedApi internal val storage: ByteArray) : Collection<UByte> {
        /** Creates a new array of the specified [size], with all elements initialized to zero. */
        public constructor(size: Int) : this(ByteArray(size))
        /** Returns the array element at the given [index]. This method can be called using the index operator. */
        operator fun get(index: Int) = storage[index].toUByte()
        /** Sets the element at the given [index] to the given [value]. This method can be called using the index operator. */
        operator fun set(index: Int, value: UByte) {
            storage[index] = value.toByte()
        }
        /** Returns the number of elements in the array. */
        public override val size: Int get() = storage.size
        /** Creates an iterator over the elements of the array. */
        public override operator fun iterator(): UByteIterator = Iterator(storage)
        private class Iterator(private val array: ByteArray) : UByteIterator() {
            private var index = 0
            override fun hasNext() = index < array.size
            override fun nextUByte() = if (index < array.size) array[index++].toUByte() else throw NoSuchElementException(index.toString())
        }
        override fun contains(element: UByte): Boolean {
            // TODO: Eliminate this check after KT-30016 gets fixed.
            @Suppress("USELESS_CAST")
            if ((element as Any?) !is UByte) return false
            return storage.contains(element.toByte())
        }
        override fun containsAll(elements: Collection<UByte>): Boolean {
            return (elements as Collection<*>).all { it is UByte && storage.contains(it.toByte()) }
        }
        override fun isEmpty(): Boolean = this.storage.size == 0
    }

    /** Creates a new array of the specified [size], where each element is calculated by calling the specified [init] function. */
    public inline fun UByteArray(size: Int, init: (Int) -> UByte): UByteArray {
        return UByteArray(ByteArray(size) { index -> init(index).toByte() })
    }

    /** Creates a new array of the specified [size], with all elements initialized to zero. */
    public constructor(size: Int) : this(ByteArray(size))

    /** Returns the array element at the given [index]. This method can be called using the index operator. */
    operator fun get(index: Int) = storage[index].toUByte()

    /** Sets the element at the given [index] to the given [value]. This method can be called using the index operator. */
    operator fun set(index: Int, value: UByte) {
        storage[index] = value.toByte()
    }

    /** Returns the number of elements in the array. */
    public override val size: Int get() = storage.size

    /** Creates an iterator over the elements of the array. */
    public override operator fun iterator(): UIntIterator = Iterator(storage)

    private class Iterator(private val array: IntArray) : UIntIterator() {
        private var index = 0
        override fun hasNext() = index < array.size
        override fun nextUInt() = if (index < array.size) array[index++].toUInt() else throw NoSuchElementException(index.toString())
    }

    override fun contains(element: UInt): Boolean {
        // TODO: Eliminate this check after KT-30016 gets fixed.
        @Suppress("USELESS_CAST")
        if ((element as Any?) !is UInt) return false
        return storage.contains(element.toUInt())
    }

    override fun containsAll(elements: Collection<UInt>): Boolean {
        return (elements as Collection<*>).all { it is UInt && storage.contains(it.toUInt()) }
    }

    override fun isEmpty(): Boolean = this.storage.size == 0
}

/*
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin
    @SinceKotlin("1.3")
    @ExperimentalUnsignedTypes
    public inline class UIntArray
            @Suppress("NON_PUBLIC_PRIMARY_CONSTRUCTOR_OF_INLINE_CLASS")
    @PublishedApi
    internal constructor(@PublishedApi internal val storage: IntArray) : Collection<UInt> {
        /** Creates a new array of the specified [size], with all elements initialized to zero. */
        public constructor(size: Int) : this(IntArray(size))
        /** Returns the array element at the given [index]. This method can be called using the index operator. */
        operator fun get(index: Int) = storage[index].toUInt()
        /** Sets the element at the given [index] to the given [value]. This method can be called using the index operator. */
        operator fun set(index: Int, value: UInt) {
            storage[index] = value.toInt()
        }
        /** Returns the number of elements in the array. */
        public override val size: Int get() = storage.size
        /** Creates an iterator over the elements of the array. */
        public override operator fun iterator(): UIntIterator = Iterator(storage)
        private class Iterator(private val array: IntArray) : UIntIterator() {
            private var index = 0
            override fun hasNext() = index < array.size
            override fun nextUInt() = if (index < array.size) array[index++].toUInt() else throw NoSuchElementException(index.toString())
        }
    }

    /** Creates a new array of the specified [size], where each element is calculated by calling the specified [init] function. */
    public inline fun UIntArray(size: Int, init: (Int) -> UInt): UIntArray {
        return UIntArray(IntArray(size) { index -> init(index).toUInt() })
    }

    /** Creates a new array of the specified [size], with all elements initialized to zero. */
    public constructor(size: Int) : this(IntArray(size))

    /** Returns the array element at the given [index]. This method can be called using the index operator. */
    operator fun get(index: Int) = storage[index].toUInt()

    /** Sets the element at the given [index] to the given [value]. This method can be called using the index operator. */
    operator fun set(index: Int, value: UInt) {
        storage[index] = value.toInt()
    }

    /** Returns the number of elements in the array. */
    public override val size: Int get() = storage.size

    /** Creates an iterator over the elements of the array. */
    public override operator fun iterator(): UIntIterator = Iterator(storage)

    private class Iterator(private val array: IntArray) : UIntIterator() {
        private var index = 0
        override fun hasNext() = index < array.size
        override fun nextUInt() = if (index < array.size) array[index++].toUInt() else throw NoSuchElementException(index.toString())
    }

    override fun contains(element: UInt): Boolean {
        // TODO: Eliminate this check after KT-30016 gets fixed.
        @Suppress("USELESS_CAST")
        if ((element as Any?) !is UInt) return false
        return storage.contains(element.toUInt())
    }

    override fun containsAll(elements: Collection<UInt>): Boolean {
        return (elements as Collection<*>).all { it is UInt && storage.contains(it.toUInt()) }
    }

    override fun isEmpty(): Boolean = this.storage.size == 0
}
contains(element: UInt): Boolean {
    // TODO: Eliminate this check after KT-30016 gets fixed.
    // Currently JS BE does not generate special bridge method for this method.
    @Suppress("USELESS_CAST")
    if ((element as Any?) !is UInt) return false
    return storage.contains(element.toInt())
}

override fun containsAll(elements: Collection<UInt>): Boolean {
    return (elements as Collection<Any>).all { it is UInt && storage.contains(it.toInt()) }
}

override fun isEmpty(): Boolean = this.storage.size == 0
}

/**
 * Creates a new array of the specified [size], where each element is calculated by calling the specified [init] function.
 * The function [init] is called for each array element sequentially starting from the first one. If it should return the value for an array element given its index.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray(size: Int, init: (Int) -> UInt): UIntArray {
    return UIntArray(IntArray(size) { index ->
        init(index).toInt()
    })
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun uintArrayOf(vararg elements: UInt): UIntArray = elements

/**
 * Creates a new array of the specified [size], where all elements initialized to zero. *\n * The constructor takes the specified [size] as a parameter.
 * It should return an array instance.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray(size: Int) :
    this(LongArray(size) { index ->
        0
    })

/**
 * Returns the array element at the given [index]. This method can be called
 * using the index operator.
 */
operator fun get(index: Int): UInt = storage[index].toUInt()

/**
 * Sets the element at the given [index] to the given [value]. This method can be called
 * using the index operator.
 */
operator fun set(index: Int, value: UInt) {
    storage[index] = value.toLong()
}

/** Returns the number of elements in the array. */
override val size: Int get() = storage.size

/** Creates an iterator over the elements of the array. */
override operator fun iterator(): UIntIterator =
    Iterator(storage)

private class Iterator(private val array: LongArray) : UIntIterator() {
    private var index = 0
    override fun hasNext() = index < array.size
    override fun nextUInt() = if (index < array.size)
        array[index++].toUInt() else throw NoSuchElementException(index.toString())
}

override fun contains(element: UInt): Boolean {
    // TODO: Eliminate this check after KT-30016 gets fixed.
    // Currently JS BE does not generate special bridge method for this method.
    @Suppress("USELESS_CAST")
    if ((element as Any?) !is UInt) return false
    return storage.contains(element.toInt())
}

override fun containsAll(elements: Collection<UInt>): Boolean {
    return (elements as Collection<*>).all { it is UInt && storage.contains(it.toInt()) }
}

override fun isEmpty(): Boolean = this.storage.size == 0
}

/**
 * Creates a new array of the specified [size], where each element is calculated by calling the specified [init] function.
 * The function [init] is called for each array element sequentially starting from the first one. If it should return the value for an array element given its index.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray(size: Int, init: (Int) -> ULong): ULongArray {
    return ULongArray(LongArray(size) { index ->
        init(index).toLong()
    })
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ulongArrayOf(vararg elements: ULong): ULongArray = elements

/**
 * Creates a new array of the specified [size], with all elements initialized to zero. *\n * The constructor takes the specified [size] as a parameter.
 * It should return an array instance.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray(size: Int) :
    this(LongArray(size) { index ->
        0
    })

/**
 * Returns the array element at the given [index]. This method can be called
 * using the index operator.
 */
operator fun get(index: Int): ULong = storage[index].toULong()

/**
 * Sets the element at the given [index] to the given [value]. This method can be called
 * using the index operator.
 */
operator fun set(index: Int, value: ULong) {
    storage[index] = value.toLong()
}

/** Returns the number of elements in the array. */
override val size: Int get() = storage.size

/** Creates an iterator over the elements of the array. */
override operator fun iterator(): ULongIterator =
    Iterator(storage)

private class Iterator(private val array: LongArray) : ULongIterator() {
    private var index = 0
    override fun hasNext() = index < array.size
    override fun nextULong() = if (index < array.size)
        array[index++].toULong() else throw NoSuchElementException(index.toString())
}

override fun contains(element: ULong): Boolean {
    // TODO: Eliminate this check after KT-30016 gets fixed.
    // Currently JS BE does not generate special bridge method for this method.
    @Suppress("USELESS_CAST")
    if ((element as Any?) !is ULong) return false
    return storage.contains(element.toLong())
}

override fun containsAll(elements: Collection<ULong>): Boolean {
    return (elements as Collection<*>).all { it is ULong && storage.contains(it.toLong()) }
}

override fun isEmpty(): Boolean = this.storage.size == 0
}

/**
 * Creates a new array of the specified [size], where each element is calculated by calling the specified [init] function.
 * The function [init] is called for each array element sequentially starting from the first one. If it should return the value for an array element given its index.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray(size: Int, init: (Int) -> ULong): ULongArray {
    return ULongArray(LongArray(size) { index ->
        init(index).toLong()
    })
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ulongArrayOf(vararg elements: ULong): ULongArray = elements

/**
 * Creates a new array of the specified [size], with all elements initialized to zero. *\n * The constructor takes the specified [size] as a parameter.
 * It should return an array instance.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray(size: Int) :
    this(LongArray(size) { index ->
        0
    })

/**
 * Returns the array element at the given [index]. This method can be called
 * using the index operator.
 */
operator fun get(index: Int): ULong = storage[index].toULong()

/**
 * Sets the element at the given [index] to the given [value]. This method can be called
 * using the index operator.
 */
operator fun set(index: Int, value: ULong) {
    storage[index] = value.toLong()
}

/** Returns the number of elements in the array. */
override val size: Int get() = storage.size

/** Creates an iterator over the elements of the array. */
override operator fun iterator(): ULongIterator =
    Iterator(storage)

private class Iterator(private val array: LongArray) : ULongIterator() {
    private var index = 0
    override fun hasNext() = index < array.size
    override fun nextULong() = if (index < array.size)
        array[index++].toULong() else throw NoSuchElementException(index.toString())
}

override fun contains(element: ULong): Boolean {
    // TODO: Eliminate this check after KT-30016 gets fixed.
    // Currently JS BE does not generate special bridge method for this method.
    @Suppress("USELESS_CAST")
    if ((element as Any?) !is ULong) return false
    return storage.contains(element.toLong())
}

override fun containsAll(elements: Collection<ULong>): Boolean {
    return (elements as Collection<*>).all { it is ULong && storage.contains(it.toLong()) }
}

override fun isEmpty(): Boolean = this.storage.size == 0
}
Ap
un internal constructor(@PublishedApi internal val storage: ShortArray) : Collection<UShort> {
    /**
     * Creates a new array of the specified [size], with all elements initialized to zero.
     * If the [index] is out of bounds of this array, throws an [IndexOutOfBounds] exception.
     */
    @PublishedApi public constructor(size: Int) : this(ShortArray(size))

    /**
     * Returns the array element at the given [index]. This method can be called using
     * the index operator.
     * * If the [index] is out of bounds of this array, throws an [IndexOutOfBounds] exception.
     * * If the [index] to the given [value]. This method can be called using the index operator.
     */
    @PublishedApi public override val size: Int get() = storage.size

    /**
     * Creates a new array of the specified [size], where each
element is calculated by calling the specified
 * 
 * init function. The function init is called for each array
element sequentially starting from the first one.
 * It should return the value for an array element given its index.
 * * Since Kotlin("1.3")\n
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
UShortArray(size: Int, init: (Int) -> UShort): UShortArray {
    return UShortArray(ShortArray(size) { index ->
init(index).toShort() })
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
ushortArrayOf(vararg elements: UShort): UShortArray = elements
}
UShortArray.component1(): UShort {\n    return get(0)\n}/**
 * Returns 2nd *element* from the array.\n * If the size of this array is less than 2, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */

*UShortArray.component2(): UShort {\n    return get(1)\n}/**
 * Returns 2nd *element* from the array.\n * If the size of this array is less than 2, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */

*UShortArray.component3(): UShort {\n    return get(2)\n}/**
 * Returns 3rd *element* from the array.\n * If the size of this array is less than 3, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */

*UShortArray.component4(): UShort {\n    return get(3)\n}/**
 * Returns 4th *element* from the array.\n * If the size of this array is less than 4, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */

*UShortArray.component5(): UShort {\n    return get(5)\n}/**
 * Returns 5th *element* from the array.\n * If the size of this array is less than 5, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */

*UIntArray.component1(): UInt {\n    return get(0)\n}/**
 * Returns 2nd *element* from the array.\n * If the size of this array is less than 2, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */

*UIntArray.component2(): UInt {\n    return get(1)\n}/**
 * Returns 2nd *element* from the array.\n * If the size of this array is less than 2, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */

*UIntArray.component3(): UInt {\n    return get(2)\n}/**
 * Returns 3rd *element* from the array.\n * If the size of this array is less than 3, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */

*UIntArray.component4(): UInt {\n    return get(3)\n}/**
 * Returns 4th *element* from the array.\n * If the size of this array is less than 4, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */

*UIntArray.component5(): UInt {\n    return get(5)\n}/**
 * Returns 5th *element* from the array.\n * If the size of this array is less than 5, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */

*ULongArray.component1(): ULong {\n    return get(0)\n}/**
 * Returns 2nd *element* from the array.\n * If the size of this array is less than 2, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */

*ULongArray.component2(): ULong {\n    return get(1)\n}/**
 * Returns 2nd *element* from the array.\n * If the size of this array is less than 2, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */

*ULongArray.component3(): ULong {\n    return get(2)\n}/**
 * Returns 3rd *element* from the array.\n * If the size of this array is less than 3, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */

*ULongArray.component4(): ULong {\n    return get(3)\n}/**
 * Returns 4th *element* from the array.\n * If the size of this array is less than 4, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */

*ULongArray.component5(): ULong {\n    return get(5)\n}/**
 * Returns 5th *element* from the array.\n * If the size of this array is less than 5, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */

*UByteArray.component1(): UByte {\n    return get(0)\n}/**
 * Returns 2nd *element* from the array.\n * If the size of this array is less than 2, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */

*UByteArray.component2(): UByte {\n    return get(1)\n}/**
 * Returns 2nd *element* from the array.\n * If the size of this array is less than 2, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */

*UByteArray.component3(): UByte {\n    return get(2)\n}/**
 * Returns 3rd *element* from the array.\n * If the size of this array is less than 3, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */

*UByteArray.component4(): UByte {\n    return get(3)\n}/**
 * Returns 4th *element* from the array.\n * If the size of this array is less than 4, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */

*UByteArray.component5(): UByte {\n    return get(5)\n}/**
 * Returns 5th *element* from the array.\n * If the size of this array is less than 5, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */

*UShortArray.component1(): UShort {\n    return get(0)\n}/**
 * Returns 2nd *element* from the array.\n * If the size of this array is less than 2, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */

*UShortArray.component2(): UShort {\n    return get(1)\n}/**
 * Returns 2nd *element* from the array.\n * If the size of this array is less than 2, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */

*UShortArray.component3(): UShort {\n    return get(2)\n}/**
 * Returns 3rd *element* from the array.\n * If the size of this array is less than 3, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */

*UShortArray.component4(): UShort {\n    return get(3)\n}/**
 * Returns 4th *element* from the array.\n * If the size of this array is less than 4, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */

*UShortArray.component5(): UShort {\n    return get(5)\n}/**
 * Returns 5th *element* from the array.\n * If the size of this array is less than 5, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */
If the size of this array is less than 5, throws an [IndexOutOfBoundsException] except in Kotlin/JS. * where the behavior is unspecified.

/*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n\npublic inline operator fun UIntArray.component5(): UInt \n    return get(4)\n\n\n*/
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun UIntArray.component5(): UInt {
    return get(4)
}

/**
 * Returns 5th *element* from the array.
 * 
 * If the size of this array is less than 5, throws an [IndexOutOfBoundsException] except in Kotlin/JS. * where the behavior is unspecified.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun ULongArray.component5(): ULong {
    return get(4)
}

/**
 * Returns 5th *element* from the array.
 * 
 * If the size of this array is less than 5, throws an [IndexOutOfBoundsException] except in Kotlin/JS. * where the behavior is unspecified.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun UByteArray.component5(): UByte {
    return get(4)
}

/**
 * Returns 5th *element* from the array.
 * 
 * If the size of this array is less than 5, throws an [IndexOutOfBoundsException] except in Kotlin/JS. * where the behavior is unspecified.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun UShortArray.component5(): UShort {
    return get(4)
}

/**
 * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 * 
 * @sample samples.collections.Collections.Elements.elementAt
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public expect fun UIntArray.elementAt(index: Int): UInt

/**
 * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 * 
 * @sample samples.collections.Collections.Elements.elementAt
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public expect fun ULongArray.elementAt(index: Int): ULong

/**
 * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 * 
 * @sample samples.collections.Collections.Elements.elementAt
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public expect fun UByteArray.elementAt(index: Int): UByte

/**
 * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 * 
 * @sample samples.collections.Collections.Elements.elementAt
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public expect fun UShortArray.elementAt(index: Int): UShort

/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.
 * 
 * @sample samples.collections.Collections.Elements.elementAtOrElse
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.elementAtOrElse(index: Int, defaultValue: (Int) -> UInt): UInt {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}

/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.
 * 
 * @sample samples.collections.Collections.Elements.elementAtOrElse
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.elementAtOrElse(index: Int, defaultValue: (Int) -> ULong): ULong {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}

/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.
 * 
 * @sample samples.collections.Collections.Elements.elementAtOrElse
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.elementAtOrElse(index: Int, defaultValue: (Int) -> UByte): UByte {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}

/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.
 * 
 * @sample samples.collections.Collections.Elements.elementAtOrElse
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.elementAtOrElse(index: Int, defaultValue: (Int) -> UShort): UShort {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}
`null` if the [index] is out of bounds of this array.

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.elementAtOrNull(index: Int): UInt? {
    return this.getOrNull(index)
}

/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 * 
 * @sample samples.collections.Collections.Elements.elementAtOrNull
 * 
 * @sample
 * 
 * @SinceKotlin("1.3")
 * @ExperimentalUnsignedTypes
 * @kotlin.internal.InlineOnly
 * public inline fun ULongArray.elementAtOrNull(index: Int): ULong? {
 *     return this.getOrNull(index)
 * }

/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 * 
 * @sample samples.collections.Collections.Elements.elementAtOrNull
 * 
 * @SinceKotlin("1.3")
 * @ExperimentalUnsignedTypes
 * @kotlin.internal.InlineOnly
 * public inline fun UByteArray.elementAtOrNull(index: Int): UByte? {
 *     return this.getOrNull(index)
 * }

/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 * 
 * @sample samples.collections.Collections.Elements.elementAtOrNull
 * 
 * @SinceKotlin("1.3")
 * @ExperimentalUnsignedTypes
 * @kotlin.internal.InlineOnly
 * public inline fun UShortArray.elementAtOrNull(index: Int): UShort? {
 *     return this.getOrNull(index)
 * }

/**
 * Returns the first element matching the given [predicate], or `null` if no such element was found.
 * 
 * @sample samples.collections.Collections.Elements.find
 * 
 * @SinceKotlin("1.3")
 * @ExperimentalUnsignedTypes
 * @kotlin.internal.InlineOnly
 * public inline fun UIntArray.find(predicate: (UInt) -> Boolean): UInt? {
 *     return firstOrNull(predicate)
 * }

/**
 * Returns the first element matching the given [predicate], or `null` if no such element was found.
 * 
 * @sample samples.collections.Collections.Elements.find
 * 
 * @SinceKotlin("1.3")
 * @ExperimentalUnsignedTypes
 * @kotlin.internal.InlineOnly
 * public inline fun ULongArray.find(predicate: (ULong) -> Boolean): ULong? {
 *     return firstOrNull(predicate)
 * }

/**
 * Returns the first element matching the given [predicate], or `null` if no such element was found.
 * 
 * @sample samples.collections.Collections.Elements.find
 * 
 * @SinceKotlin("1.3")
 * @ExperimentalUnsignedTypes
 * @kotlin.internal.InlineOnly
 * public inline fun UByteArray.find(predicate: (UByte) -> Boolean): UByte? {
 *     return firstOrNull(predicate)
 * }

/**
 * Returns the first element matching the given [predicate], or `null` if no such element was found.
 * 
 * @sample samples.collections.Collections.Elements.find
 * 
 * @SinceKotlin("1.3")
 * @ExperimentalUnsignedTypes
 * @kotlin.internal.InlineOnly
 * public inline fun UShortArray.find(predicate: (UShort) -> Boolean): UShort? {
 *     return firstOrNull(predicate)
 * }

/**
 * Returns the last element matching the given [predicate], or `null` if no such element was found.
 * 
 * @sample samples.collections.Collections.Elements.findLast
 * 
 * @SinceKotlin("1.3")
 * @ExperimentalUnsignedTypes
 * @kotlin.internal.InlineOnly
 * public inline fun UIntArray.findLast(predicate: (UInt) -> Boolean): UInt? {
 *     return lastOrNull(predicate)
 * }

/**
 * Returns the last element matching the given [predicate], or `null` if no such element was found.
 * 
 * @sample samples.collections.Collections.Elements.findLast
 * 
 * @SinceKotlin("1.3")
 * @ExperimentalUnsignedTypes
 * @kotlin.internal.InlineOnly
 * public inline fun ULongArray.findLast(predicate: (ULong) -> Boolean): ULong? {
 *     return lastOrNull(predicate)
 * }

/**
 * Returns the last element matching the given [predicate], or `null` if no such element was found.
 * 
 * @sample samples.collections.Collections.Elements.findLast
 * 
 * @SinceKotlin("1.3")
 * @ExperimentalUnsignedTypes
 * @kotlin.internal.InlineOnly
 * public inline fun UByteArray.findLast(predicate: (UByte) -> Boolean): UByte? {
 *     return lastOrNull(predicate)
 * }

/**
 * Returns the last element matching the given [predicate], or `null` if no such element was found.
 * 
 * @sample samples.collections.Collections.Elements.findLast
 * 
 * @SinceKotlin("1.3")
 * @ExperimentalUnsignedTypes
 * @kotlin.internal.InlineOnly
 * public inline fun UShortArray.findLast(predicate: (UShort) -> Boolean): UShort? {
 *     return lastOrNull(predicate)
 * }

```
Returns first element.

*Throws [NoSuchElementException] if the array is empty.*

```kotlin
*Throws [NoSuchElementException] if the array is empty.*

```
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.firstOrNull(predicate: (UByte) -> Boolean): UByte? {
    for (element in this) if (predicate(element))
        return element
    return null
}

/**
 * Returns the first element matching the given [predicate], or `null` if
 * element was not found.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.firstOrNull(predicate: (UShort) -> Boolean): UShort? {
    for (element in this) if (predicate(element))
        return element
    return null
}

/**
 * Returns an element at the given [index] or the result of calling the
 * [defaultValue] function if the [index] is out of bounds of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.getOrElse(index: Int, defaultValue: (Int) -> UInt): UInt {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}

/**
 * Returns an element at the given [index] or `null` if the
 * [index] is out of bounds of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.getOrElse(index: Int, defaultValue: (Int) -> ULong): ULong {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}

/**
 * Returns first index of
 * [element], or -1 if the array does not contain element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.indexOf(element: UInt): Int {
    return storage.indexOf(element.toInt())
}

/**
 * Returns first
 * index of 
 * [element], or -1 if the array does not contain element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.indexOf(element: ULong): Int {
    return storage.indexOf(element.toLong())
}

/**
 * Returns first index of 
 * [element], or -1 if the array does not contain element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.indexOf(element: UByte): Int {
    return storage.indexOf(element.toByte())
}

/**
 * Returns first index of 
 * [element], or -1 if the array does not contain element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.indexOf(element: UShort): Int {
    return storage.indexOf(element)
*\@SinceKotlin("1.3")*\@ExperimentalUnsignedTypes*\@kotlin.internal.InlineOnly*\public inline fun UShortArray.indexOf(element: UShort): Int {\n    return storage.indexOf(element.toShort())\n}\n/**
 * Returns index of the first element matching the given [predicate], or -1 if the array does not contain such element.
 */
*\@SinceKotlin("1.3")*\@ExperimentalUnsignedTypes*\@kotlin.internal.InlineOnly*\public inline fun UIntArray.indexOfFirst(predicate: (UInt) -> Boolean): Int {\n    return storage.indexOfFirst { predicate(it.toUInt()) }\n}\n/**
 * Returns index of the first element matching the given [predicate], or -1 if the array does not contain such element.
 */
*\@SinceKotlin("1.3")*\@ExperimentalUnsignedTypes*\@kotlin.internal.InlineOnly*\public inline fun ULongArray.indexOfFirst(predicate: (ULong) -> Boolean): Int {\n    return storage.indexOfFirst { predicate(it.toULong()) }\n}\n/**
 * Returns index of the first element matching the given [predicate], or -1 if the array does not contain such element.
 */
*\@SinceKotlin("1.3")*\@ExperimentalUnsignedTypes*\@kotlin.internal.InlineOnly*\public inline fun UByteArray.indexOfFirst(predicate: (UByte) -> Boolean): Int {\n    return storage.indexOfFirst { predicate(it.toUByte()) }\n}\n/**
 * Returns index of the first element matching the given [predicate], or -1 if the array does not contain such element.
 */
*\@SinceKotlin("1.3")*\@ExperimentalUnsignedTypes*\@kotlin.internal.InlineOnly*\public inline fun UShortArray.indexOfLast(predicate: (UShort) -> Boolean): Int {\n    return storage.indexOfLast { predicate(it.toUShort()) }\n}\n/**
 * Returns index of the last element matching the given [predicate], or -1 if the array does not contain such element.
 */
*\@SinceKotlin("1.3")*\@ExperimentalUnsignedTypes*\@kotlin.internal.InlineOnly*\public inline fun UIntArray.indexOfLast(predicate: (UInt) -> Boolean): Int {\n    return storage.indexOfLast { predicate(it.toUInt()) }\n}\n/**
 * Returns index of the last element matching the given [predicate], or -1 if the array does not contain such element.
 */
*\@SinceKotlin("1.3")*\@ExperimentalUnsignedTypes*\@kotlin.internal.InlineOnly*\public inline fun ULongArray.indexOfLast(predicate: (ULong) -> Boolean): Int {\n    return storage.indexOfLast { predicate(it.toULong()) }\n}\n/**
 * Returns index of the last element matching the given [predicate], or -1 if the array does not contain such element.
 */
*\@SinceKotlin("1.3")*\@ExperimentalUnsignedTypes*\@kotlin.internal.InlineOnly*\public inline fun UByteArray.indexOfLast(predicate: (UByte) -> Boolean): Int {\n    return storage.indexOfLast { predicate(it.toUByte()) }\n}\n/**
 * Returns index of the last element matching the given [predicate], or -1 if the array does not contain such element.
 */
*\@SinceKotlin("1.3")*\@ExperimentalUnsignedTypes*\@kotlin.internal.InlineOnly*\public inline fun UShortArray.indexOfLast(predicate: (UShort) -> Boolean): Int {\n    return storage.indexOfLast { predicate(it.toUShort()) }\n}\n/**
 * Returns the last element.
 * @throws NoSuchElementException if the array is empty.
 * @sample samples.collections.Elements.last
 */
*\@SinceKotlin("1.3")*\@ExperimentalUnsignedTypes*\@kotlin.internal.InlineOnly*\public inline fun UIntArray.last(): UInt {\n    return storage.last().toUInt()\n}\n/**
 * Returns the last element.
 * @throws NoSuchElementException if the array is empty.
 * @sample samples.collections.Elements.last
 */
*\@SinceKotlin("1.3")*\@ExperimentalUnsignedTypes*\@kotlin.internal.InlineOnly*\public inline fun ULongArray.last(): ULong {\n    return storage.last().toULong()\n}\n/**
 * Returns the last element.
 * @throws NoSuchElementException if the array is empty.
 * @sample samples.collections.Elements.last
 */
*\@SinceKotlin("1.3")*\@ExperimentalUnsignedTypes*\@kotlin.internal.InlineOnly*\public inline fun UByteArray.last(): UByte {\n    return storage.last().toUByte()\n}\n/**
 * Returns the last element.
 * @throws NoSuchElementException if the array is empty.
 * @sample samples.collections.Elements.last
 */
*\@SinceKotlin("1.3")*\@ExperimentalUnsignedTypes*\@kotlin.internal.InlineOnly*\public inline fun UShortArray.last(): UShort {\n    return storage.last().toUShort()\n}
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
UIntArray.last(predicate: (UInt) -> Boolean): UInt {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns the last element matching the given [predicate].
 * 
 * @throws NoSuchElementException if no such element is found.
 * 
 * @sample samples.collections.Collections.Elements.last
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
ULongArray.last(predicate: (ULong) -> Boolean): ULong {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns the last element matching the given [predicate].
 * 
 * @throws NoSuchElementException if no such element is found.
 * 
 * @sample samples.collections.Collections.Elements.last
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
UByteArray.last(predicate: (UByte) -> Boolean): UByte {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns the last element matching the given [predicate].
 * 
 * @throws NoSuchElementException if no such element is found.
 * 
 * @sample samples.collections.Collections.Elements.last
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
UShortArray.last(predicate: (UShort) -> Boolean): UShort {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns last index of [element], or -1 if the array does not contain element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
UIntArray.lastIndexOf(element: UInt): Int {
    return storage.lastIndexOf(element.toInt())
}

/**
 * Returns last index of [element], or -1 if the array does not contain element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
ULongArray.lastIndexOf(element: ULong): Int {
    return storage.lastIndexOf(element.toLong())
}

/**
 * Returns last index of [element], or -1 if the array does not contain element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
UByteArray.lastIndexOf(element: UByte): Int {
    return storage.lastIndexOf(element.toByte())
}

/**
 * Returns last index of [element], or -1 if the array does not contain element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
UShortArray.lastIndexOf(element: UShort): Int {
    return storage.lastIndexOf(element.toShort())
}

/**
 * Returns the last element, or `null` if the array is empty.
 * 
 * @sample samples.collections.Collections.Elements.last
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray.lastOrNull(): UInt? {
    return if (isEmpty()) null else this[size - 1]
}

/**
 * Returns the last element, or `null` if the array is empty.
 * 
 * @sample samples.collections.Collections.Elements.last
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongArray.lastOrNull(): ULong? {
    return if (isEmpty()) null else this[size - 1]
}

/**
 * Returns the last element, or `null` if the array is empty.
 * 
 * @sample samples.collections.Collections.Elements.last
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.lastOrNull(): UByte? {
    return if (isEmpty()) null else this[size - 1]
}

/**
 * Returns the last element, or `null` if the array is empty.
 * 
 * @sample samples.collections.Collections.Elements.last
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.lastOrNull(): UShort? {
    return if (isEmpty()) null else this[size - 1]
}

/**
 * Returns the last element matching the given [predicate], or `null` if no such element was found.
 * 
 * @sample samples.collections.Collections.Elements.last
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.lastOrNull(): UByte? {
    return if (isEmpty()) null else this[size - 1]
}

/**
 * Returns the last element matching the given [predicate], or `null` if no such element was found.
 * 
 * @sample samples.collections.Collections.Elements.last
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.lastOrNull(): UShort? {
    return if (isEmpty()) null else this[size - 1]
}

/**
 * Returns the last element matching the given [predicate], or `null` if no such element was found.
 * 
 * @sample samples.collections.Collections.Elements.last
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.lastOrNull(): UByte? {
    return if (isEmpty()) null else this[size - 1]
}

/**
 * Returns the last element matching the given [predicate], or `null` if no such element was found.
 * 
 * @sample samples.collections.Collections.Elements.last
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.lastOrNull(): UShort? {
    return if (isEmpty()) null else this[size - 1]
}
UIntArray.lastOrNull(predicate: (UInt) -> Boolean): UInt? { 
    for (index in this.indices.reversed()) { 
        if (predicate(element)) return element
    }
    return null
}

/**
 * Returns the last element matching the given [predicate], or `null` if no such element was found.
 * 
 * @throws NoSuchElementException if this array is empty.
 * 
 * @sample samples.collections.Collections.Elements.last
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.lastOrNull(predicate: (ULong) -> Boolean): ULong? {
    for (index in this.indices.reversed()) { 
        if (predicate(element)) return element
    }
    return null
}

/**
 * Returns the last element matching the given [predicate], or `null` if no such element was found.
 * 
 * @throws NoSuchElementException if this array is empty.
 * 
 * @sample samples.collections.Collections.Elements.last
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.lastOrNull(predicate: (UByte) -> Boolean): UByte? {
    for (index in this.indices.reversed()) { 
        if (predicate(element)) return element
    }
    return null
}

/**
 * Returns the last element matching the given [predicate], or `null` if no such element was found.
 * 
 * @throws NoSuchElementException if this array is empty.
 * 
 * @sample samples.collections.Collections.Elements.last
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.lastOrNull(predicate: (UShort) -> Boolean): UShort? {
    for (index in this.indices.reversed()) { 
        if (predicate(element)) return element
    }
    return null
}

/**
 * Returns a random element from this array.
 * 
 * @throws NoSuchElementException if this array is empty.
 * 
 * @sample samples.collections.Collections.Elements.random
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.random(): UInt {
    return random(Random)
}

/**
 * Returns a random element from this array.
 * 
 * @throws NoSuchElementException if this array is empty.
 * 
 * @sample samples.collections.Collections.Elements.random
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.random(): ULong {
    return random(Random)
}

/**
 * Returns a random element from this array.
 * 
 * @throws NoSuchElementException if this array is empty.
 * 
 * @sample samples.collections.Collections.Elements.random
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.random(): UByte {
    return random(Random)
}

/**
 * Returns a random element from this array.
 * 
 * @throws NoSuchElementException if this array is empty.
 * 
 * @sample samples.collections.Collections.Elements.random
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.random(): UShort {
    return random(Random)
}

/**
 * Returns a random element from this array using the specified source of randomness.
 * 
 * @throws NoSuchElementException if this array is empty.
 * 
 * @sample samples.collections.Collections.Elements.random
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.randomOrNull(): UInt? {
    return randomOrNull(Random)
}

/**
 * Returns a random element from this array using the specified source of randomness.
 * 
 * @throws NoSuchElementException if this array is empty.
 * 
 * @sample samples.collections.Collections.Elements.random
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.randomOrNull(): ULong? {
    return randomOrNull(Random)
}

/**
 * Returns a random element from this array using the specified source of randomness.
 * 
 * @throws NoSuchElementException if this array is empty.
 * 
 * @sample samples.collections.Collections.Elements.random
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.randomOrNull(): UByte? {
    return randomOrNull(Random)
}

/**
 * Returns a random element from this array using the specified source of randomness.
 * 
 * @throws NoSuchElementException if this array is empty.
 * 
 * @sample samples.collections.Collections.Elements.random
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.randomOrNull(): UShort? {
    return randomOrNull(Random)
}
@kotlin.internal.InlineOnly
public inline fun ULongArray.randomOrNull(): ULong? {
    return randomOrNull(Random)
}

/**
 * Returns a random element from this array, or `null` if this array is empty.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes

@kotlin.internal.InlineOnly
public inline fun UByteArray.randomOrNull(): UByte? {
    return randomOrNull(Random)
}

/**
 * Returns a random element from this array, or `null` if this array is empty.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes

@kotlin.internal.InlineOnly
public inline fun UShortArray.randomOrNull(): UShort? {
    return randomOrNull(Random)
}

/**
 * Returns a random element from this array using the specified source of randomness, or `null` if this array is empty.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes

public fun UIntArray.randomOrNull(random: Random): UInt? {
    if (isEmpty())
        return null
    return get(random.nextInt(size))
}

/**
 * Returns a random element from this array using the specified source of randomness, or `null` if this array is empty.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes

public fun ULongArray.randomOrNull(random: Random): ULong? {
    if (isEmpty())
        return null
    return get(random.nextInt(size))
}

/**
 * Returns a random element from this array using the specified source of randomness, or `null` if this array is empty.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes

public fun UByteArray.randomOrNull(random: Random): UByte? {
    if (isEmpty())
        return null
    return get(random.nextInt(size))
}

/**
 * Returns a random element from this array using the specified source of randomness, or `null` if this array is empty.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes

public fun UShortArray.randomOrNull(random: Random): UShort? {
    if (isEmpty())
        return null
    return get(random.nextInt(size))
}

/**
 * Returns the single element, or throws an exception if the array is empty or has more than one element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.single(): UInt {
    return storage.single().toUInt()
}

/**
 * Returns the single element, or throws an exception if the array is empty or has more than one element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.single(): ULong {
    return storage.single().toULong()
}

/**
 * Returns the single element, or throws an exception if the array is empty or has more than one element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.single(): UByte {
    return storage.single().toUByte()
}

/**
 * Returns the single element, or throws an exception if the array is empty or has more than one element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.single(): UShort {
    return storage.single().toUShort()
}

/**
 * Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.single(predicate: (UInt) -> Boolean): UInt {
    var single: UInt? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")

    @Suppress("UNCHECKED_CAST")
    return single as UInt
}

/**
 * Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.single(predicate: (ULong) -> Boolean): ULong {
    var single: ULong? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")

    @Suppress("UNCHECKED_CAST")
    return single as ULong
}
contains more than one matching element.

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.singleOrNull(): UByte? {
    return if (size == 1) this[0] else null
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.singleOrNull(predicate: (UInt) -> Boolean):(UInt? {
    var single: UInt? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) return null
            single = element
            found = true
        }
    }
    if (!found) return null
    return single
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.singleOrNull(predicate: (UShort) -> Boolean): UShort? {
    var single: UShort? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) return null
            single = element
            found = true
        }
    }
    if (!found) return null
    return single
}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.singleOrNull(): UByte? {
    return if (size == 1) this[0] else null
}
```
fun UIntArray.drop(n: Int): List<UInt> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return takeLast((size - n).coerceAtLeast(0))
}

/**
 * Returns a list containing all elements except first [n] elements.
 * 
 * @throws IllegalArgumentException if [n] is negative.
 * 
 * @sample samples.collections.Collections.Transformations.drop
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongArray.drop(n: Int): List<ULong> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return takeLast((size - n).coerceAtLeast(0))
}

/**
 * Returns a list containing all elements except first [n] elements.
 * 
 * @throws IllegalArgumentException if [n] is negative.
 * 
 * @sample samples.collections.Collections.Transformations.drop
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.drop(n: Int): List<UByte> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return takeLast((size - n).coerceAtLeast(0))
}

/**
 * Returns a list containing all elements except first [n] elements.
 * 
 * @throws IllegalArgumentException if [n] is negative.
 * 
 * @sample samples.collections.Collections.Transformations.drop
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.drop(n: Int): List<UShort> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return takeLast((size - n).coerceAtLeast(0))
}

/**
 * Returns a list containing all elements except last [n] elements.
 * 
 * @throws IllegalArgumentException if [n] is negative.
 * 
 * @sample samples.collections.Collections.Transformations.drop
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray.dropLast(n: Int): List<UInt> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return take((size - n).coerceAtLeast(0))
}

/**
 * Returns a list containing all elements except last [n] elements.
 * 
 * @throws IllegalArgumentException if [n] is negative.
 * 
 * @sample samples.collections.Collections.Transformations.drop
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongArray.dropLast(n: Int): List<ULong> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return take((size - n).coerceAtLeast(0))
}

/**
 * Returns a list containing all elements except last [n] elements.
 * 
 * @throws IllegalArgumentException if [n] is negative.
 * 
 * @sample samples.collections.Collections.Transformations.drop
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.dropLast(n: Int): List<UByte> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return take((size - n).coerceAtLeast(0))
}

/**
 * Returns a list containing all elements except last [n] elements.
 * 
 * @throws IllegalArgumentException if [n] is negative.
 * 
 * @sample samples.collections.Collections.Transformations.drop
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.dropLast(n: Int): List<UShort> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return take((size - n).coerceAtLeast(0))
}

/**
 * Returns a list containing all elements except last elements that satisfy the given [predicate].
 * 
 * @throws IllegalArgumentException if [n] is negative.
 * 
 * @sample samples.collections.Collections.Transformations.drop
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.dropLastWhile(predicate: (UInt) -> Boolean): List<UInt> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return take(index + 1)
        }
    }
    return emptyList()
}

/**
 * Returns a list containing all elements except last elements that satisfy the given [predicate].
 * 
 * @throws IllegalArgumentException if [n] is negative.
 * 
 * @sample samples.collections.Collections.Transformations.drop
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.dropLastWhile(predicate: (ULong) -> Boolean): List<ULong> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return take(index + 1)
        }
    }
    return emptyList()
}

/**
 * Returns a list containing all elements except last elements that satisfy the given [predicate].
 * 
 * @throws IllegalArgumentException if [n] is negative.
 * 
 * @sample samples.collections.Collections.Transformations.drop
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.dropLastWhile(predicate: (UByte) -> Boolean): List<UByte> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return take(index + 1)
        }
    }
    return emptyList()
}

/**
 * Returns a list containing all elements except last elements that satisfy the given [predicate].
 * 
 * @throws IllegalArgumentException if [n] is negative.
 * 
 * @sample samples.collections.Collections.Transformations.drop
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.dropLastWhile(predicate: (UShort) -> Boolean): List<UShort> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return take(index + 1)
        }
    }
    return emptyList()
* Returns a list containing all elements except last elements that satisfy the given [predicate].

```kotlin
fun UByteArray.dropLastWhile(predicate: (UByte) -> Boolean): List<UByte> {
    var yielding = false
    val list = ArrayList<UByte>()
    for (item in this) {
        if (yielding)
            list.add(item)
        else if (!predicate(item)) {
            list.add(item)
            yielding = true
        }
    }
    return list
}
```

```kotlin
fun UShortArray.dropLastWhile(predicate: (UShort) -> Boolean): List<UShort> {
    var yielding = false
    val list = ArrayList<UShort>()
    for (item in this) {
        if (yielding)
            list.add(item)
        else if (!predicate(item)) {
            list.add(item)
            yielding = true
        }
    }
    return list
}
```

```kotlin
fun UIntArray.dropWhile(predicate: (UInt) -> Boolean): List<UInt> {
    var yielding = false
    val list = ArrayList<UInt>()
    for (item in this) {
        if (yielding)
            list.add(item)
        else if (!predicate(item)) {
            list.add(item)
            yielding = true
        }
    }
    return list
}
```

```kotlin
fun ULongArray.dropWhile(predicate: (ULong) -> Boolean): List<ULong> {
    var yielding = false
    val list = ArrayList<ULong>()
    for (item in this) {
        if (yielding)
            list.add(item)
        else if (!predicate(item)) {
            list.add(item)
            yielding = true
        }
    }
    return list
}
```

```kotlin
fun UByteArray.dropWhile(predicate: (UByte) -> Boolean): List<UByte> {
    var yielding = false
    val list = ArrayList<UByte>()
    for (item in this) {
        if (yielding)
            list.add(item)
        else if (!predicate(item)) {
            list.add(item)
            yielding = true
        }
    }
    return list
}
```

```kotlin
fun UShortArray.dropWhile(predicate: (UShort) -> Boolean): List<UShort> {
    var yielding = false
    val list = ArrayList<UShort>()
    for (item in this) {
        if (yielding)
            list.add(item)
        else if (!predicate(item)) {
            list.add(item)
            yielding = true
        }
    }
    return list
}
```

```kotlin
fun UIntArray.filter(predicate: (UInt) -> Boolean): List<UInt> {
    return filterTo(ArrayList<UInt>(), predicate)
}
```

```kotlin
fun ULongArray.filter(predicate: (ULong) -> Boolean): List<ULong> {
    return filterTo(ArrayList<ULong>(), predicate)
}
```

```kotlin
fun UByteArray.filter(predicate: (UByte) -> Boolean): List<UByte> {
    return filterTo(ArrayList<UByte>(), predicate)
}
```

```kotlin
fun UShortArray.filter(predicate: (UShort) -> Boolean): List<UShort> {
    return filterTo(ArrayList<UShort>(), predicate)
}
```

```kotlin
fun UIntArray.filter(predicate: (UInt) -> Boolean): List<UInt> {
    return filterTo(ArrayList<UInt>(), predicate)
}
```

```kotlin
fun ULongArray.filter(predicate: (ULong) -> Boolean): List<ULong> {
    return filterTo(ArrayList<ULong>(), predicate)
}
```

```kotlin
fun UByteArray.filter(predicate: (UByte) -> Boolean): List<UByte> {
    return filterTo(ArrayList<UByte>(), predicate)
}
```

```kotlin
fun UShortArray.filter(predicate: (UShort) -> Boolean): List<UShort> {
    return filterTo(ArrayList<UShort>(), predicate)
}
```

```kotlin
fun UIntArray.filter(predicate: (UInt) -> Boolean): List<UInt> {
    return filterTo(ArrayList<UInt>(), predicate)
}
```

```kotlin
fun ULongArray.filter(predicate: (ULong) -> Boolean): List<ULong> {
    return filterTo(ArrayList<ULong>(), predicate)
}
```

```kotlin
fun UByteArray.filter(predicate: (UByte) -> Boolean): List<UByte> {
    return filterTo(ArrayList<UByte>(), predicate)
}
```

```kotlin
fun UShortArray.filter(predicate: (UShort) -> Boolean): List<UShort> {
    return filterTo(ArrayList<UShort>(), predicate)
}
```

```kotlin
fun UIntArray.filter(predicate: (UInt) -> Boolean): List<UInt> {
    return filterTo(ArrayList<UInt>(), predicate)
}
```

```kotlin
fun ULongArray.filter(predicate: (ULong) -> Boolean): List<ULong> {
    return filterTo(ArrayList<ULong>(), predicate)
}
```

```kotlin
fun UByteArray.filter(predicate: (UByte) -> Boolean): List<UByte> {
    return filterTo(ArrayList<UByte>(), predicate)
}
```

```kotlin
fun UShortArray.filter(predicate: (UShort) -> Boolean): List<UShort> {
    return filterTo(ArrayList<UShort>(), predicate)
}
```
UShortArray.filter(predicate: (UShort) -> Boolean): List<UShort> {
    return filterTo(ArrayList<UShort>(), predicate)
}

/**
 * Returns a list containing only elements matching the given [predicate].
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 * @sample samples.collections.Collections.Filtering.filter
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.filterIndexed(predicate: (index: Int, UShort) -> Boolean): List<UShort> {
    return filterIndexedTo(ArrayList<UShort>(), predicate)
}

/**
 * Returns a list containing only elements matching the given [predicate].
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 * @sample samples.collections.Collections.Filtering.filterIndexed
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.filterIndexed(predicate: (index: Int, UInt) -> Boolean): List<UInt> {
    return filterIndexedTo(ArrayList<UInt>(), predicate)
}

/**
 * Returns a list containing only elements matching the given [predicate].
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 * @sample samples.collections.Collections.Filtering.filterIndexed
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.filterIndexed(predicate: (index: Int, ULong) -> Boolean): List<ULong> {
    return filterIndexedTo(ArrayList<ULong>(), predicate)
}

/**
 * Returns a list containing only elements matching the given [predicate].
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 * @sample samples.collections.Collections.Filtering.filterIndexed
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.filterIndexed(predicate: (index: Int, UByte) -> Boolean): List<UByte> {
    return filterIndexedTo(ArrayList<UByte>(), predicate)
}

/**
 * Returns a list containing only elements matching the given [predicate].
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 * @sample samples.collections.Collections.Filtering.filterIndexed
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.filterIndexed(predicate: (index: Int, UShort) -> Boolean): List<UShort> {
    return filterIndexedTo(ArrayList<UShort>(), predicate)
}

/**
 * Appends all elements matching the given [predicate] to the given [destination]
 * and returns the result of predicate evaluation on the element.
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 * @sample samples.collections.Collections.Filtering.filterIndexedTo
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <C : MutableCollection<in UInt>> UIntArray.filterIndexedTo(destination: C, predicate: (index: Int, UInt) -> Boolean): C {
    forEachIndexed { index, element ->
        if (predicate(index, element)) destination.add(element)
    }
    return destination
}

/**
 * Appends all elements matching the given [predicate] to the given [destination]
 * and returns the result of predicate evaluation on the element.
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 * @sample samples.collections.Collections.Filtering.filterIndexedTo
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <C : MutableCollection<in ULong>> ULongArray.filterIndexedTo(destination: C, predicate: (index: Int, ULong) -> Boolean): C {
    forEachIndexed { index, element ->
        if (predicate(index, element)) destination.add(element)
    }
    return destination
}

/**
 * Appends all elements matching the given [predicate] to the given [destination]
 * and returns the result of predicate evaluation on the element.
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 * @sample samples.collections.Collections.Filtering.filterIndexedTo
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <C : MutableCollection<in UByte>> UByteArray.filterIndexedTo(destination: C, predicate: (index: Int, UByte) -> Boolean): C {
    forEachIndexed { index, element ->
        if (predicate(index, element)) destination.add(element)
    }
    return destination
}

/**
 * Appends all elements matching the given [predicate] to the given [destination]
 * and returns the result of predicate evaluation on the element.
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 * @sample samples.collections.Collections.Filtering.filterIndexedTo
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <C : MutableCollection<in UShort>> UShortArray.filterIndexedTo(destination: C, predicate: (index: Int, UShort) -> Boolean): C {
    forEachIndexed { index, element ->
        if (predicate(index, element)) destination.add(element)
    }
    return destination
}

/**
 * Appends all elements matching the given [predicate] to the given [destination]
 * and returns the result of predicate evaluation on the element.
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 * @sample samples.collections.Collections.Filtering.filterIndexedTo
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <C : MutableCollection<in UByte>> UByteArray.filterIndexedTo(destination: C, predicate: (index: Int, UByte) -> Boolean): C {
    forEachIndexed { index, element ->
        if (predicate(index, element)) destination.add(element)
    }
    return destination
}

**Open Source Used In security_automation identity-guest-cs 1497**
/*
 * Since Kotlin("1.3") at ExperimentalUnsignedTypes in kotlin.internal.InlineOnly
 * public inline fun <C : MutableCollection<in UShort>> UShortArray.filterIndexedTo(destination: C, predicate: (Int, UShort) -> Boolean): C {
 * forEachIndexed { index, element ->
 * if (predicate(index, element))
 * destination.add(element)
 * return destination
 * } * Returns a list containing all elements not matching the given [predicate].
 * @sample samples.collections.Collections.Filtering.filter
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <C : MutableCollection<in ULong>> ULongArray.filterIndexedTo(destination: C, predicate: (ULong) -> Boolean): List<ULong> {
 * for (element in this) if (!predicate(element)) destination.add(element)
 * return destination
 * } * Returns a list containing all elements not matching the given [predicate].
 * @sample samples.collections.Collections.Filtering.filter

/*
 * Returns a list containing all elements not matching the given [predicate].
 * @sample samples.collections.Collections.Filtering.filter
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.filterNot(predicate: (UInt) -> Boolean): List<UInt> {
 * return filterNotTo(ArrayList<UInt>(), predicate)
 * } * Returns a list containing all elements not matching the given [predicate].
 * @sample samples.collections.Collections.Filtering.filter

/*
 * Returns a list containing all elements not matching the given [predicate].
 * @sample samples.collections.Collections.Filtering.filter
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.filterNot(predicate: (ULong) -> Boolean): List<ULong> {
 * return filterNotTo(ArrayList<ULong>(), predicate)
 * } * Returns a list containing all elements not matching the given [predicate].
 * @sample samples.collections.Collections.Filtering.filter

/*
 * Returns a list containing all elements not matching the given [predicate].
 * @sample samples.collections.Collections.Filtering.filter
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.filterNot(predicate: (UByte) -> Boolean): List<UByte> {
 * return filterNotTo(ArrayList<UByte>(), predicate)
 * } * Returns a list containing all elements not matching the given [predicate].
 * @sample samples.collections.Collections.Filtering.filter

/*
 * Returns a list containing all elements not matching the given [predicate].
 * @sample samples.collections.Collections.Filtering.filter
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.filterNot(predicate: (UShort) -> Boolean): List<UShort> {
 * return filterNotTo(ArrayList<UShort>(), predicate)
 * } * Returns a list containing all elements not matching the given [predicate].
 * @sample samples.collections.Collections.Filtering.filter

/*
 * Appends all elements not matching the given [predicate] to the given [destination].
 * @sample samples.collections.Collections.Filtering.filterTo
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <C : MutableCollection<in UInt>> UIntArray.filterNotTo(destination: C, predicate: (UInt) -> Boolean): C {
 * for (element in this) if (!predicate(element)) destination.add(element)
 * return destination
 * } * Appends all elements not matching the given [predicate] to the given [destination].
 * @sample samples.collections.Collections.Filtering.filterTo

/*
 * Appends all elements not matching the given [predicate] to the given [destination].
 * @sample samples.collections.Collections.Filtering.filterTo
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <C : MutableCollection<in ULong>> ULongArray.filterNotTo(destination: C, predicate: (ULong) -> Boolean): C {
 * for (element in this) if (!predicate(element)) destination.add(element)
 * return destination
 * } * Appends all elements not matching the given [predicate] to the given [destination].
 * @sample samples.collections.Collections.Filtering.filterTo

/*
 * Appends all elements not matching the given [predicate] to the given [destination].
 * @sample samples.collections.Collections.Filtering.filterTo
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <C : MutableCollection<in UByte>> UByteArray.filterNotTo(destination: C, predicate: (UByte) -> Boolean): C {
 * for (element in this) if (!predicate(element)) destination.add(element)
 * return destination
 * } * Appends all elements not matching the given [predicate] to the given [destination].
 * @sample samples.collections.Collections.Filtering.filterTo

/*
 * Appends all elements not matching the given [predicate] to the given [destination].
 * @sample samples.collections.Collections.Filtering.filterTo
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <C : MutableCollection<in UShort>> UShortArray.filterNotTo(destination: C, predicate: (UShort) -> Boolean): C {
 * for (element in this) if (!predicate(element)) destination.add(element)
 * return destination
 * } * Appends all elements not matching the given [predicate] to the given [destination].
 * @sample samples.collections.Collections.Filtering.filterTo

/*
 * Appends all elements matching the given [predicate] to the given [destination].
 * @sample samples.collections.Collections.Filtering.filterTo
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <C : MutableCollection<in UInt>> UIntArray.filterTo(destination: C, predicate: (UInt) -> Boolean): C {
 * for (element in this) if (predicate(element)) destination.add(element)
 * return destination
 * } * Appends all elements matching the given [predicate] to the given [destination].
 * @sample samples.collections.Collections.Filtering.filterTo

/*
 * Appends all elements matching the given [predicate] to the given [destination].
 * @sample samples.collections.Collections.Filtering.filterTo
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <C : MutableCollection<in ULong>> ULongArray.filterTo(destination: C, predicate: (ULong) -> Boolean): C {
 * for (element in this) if (predicate(element)) destination.add(element)
 * return destination
 * } * Appends all elements matching the given [predicate] to the given [destination].
 * @sample samples.collections.Collections.Filtering.filterTo

/*
 * Appends all elements matching the given [predicate] to the given [destination].
 * @sample samples.collections.Collections.Filtering.filterTo
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <C : MutableCollection<in UByte>> UByteArray.filterTo(destination: C, predicate: (UByte) -> Boolean): C {
 * for (element in this) if (predicate(element)) destination.add(element)
 * return destination
 * } * Appends all elements matching the given [predicate] to the given [destination].
 * @sample samples.collections.Collections.Filtering.filterTo

/*
 * Appends all elements matching the given [predicate] to the given [destination].
 * @sample samples.collections.Collections.Filtering.filterTo
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <C : MutableCollection<in UShort>> UShortArray.filterTo(destination: C, predicate: (UShort) -> Boolean): C {
 * for (element in this) if (predicate(element)) destination.add(element)
 * return destination
 * } * Appends all elements matching the given [predicate] to the given [destination].
 * @sample samples.collections.Collections.Filtering.filterTo

Open Source Used in security_automation identity-guest-cs 1498
(element in this) if (predicate(element)) destination.add(element) return destination
} * Appends all elements matching the given [predicate] to the given [destination].

* @sample samples.collections.Collections.Filtering.filterTo

* @SinceKotlin("1.3") @ExperimentalUnsignedTypes

fun UByteArray.slice(indices: IntRange): List<UByte>
{
    val size = indices.collectionSizeOrDefault(10)
    if (size == 0) return emptyList()
    val list = ArrayList<UByte>(size)
    for (index in indices) { list.add(get(index)) }
    return list
}

* @SinceKotlin("1.3") @ExperimentalUnsignedTypes

fun UShortArray.slice(indices: IntRange): List<UShort>
{
    val size = indices.collectionSizeOrDefault(10)
    if (size == 0) return emptyList()
    val list = ArrayList<UShort>(size)
    for (index in indices) { list.add(get(index)) }
    return list
}

* @SinceKotlin("1.3") @ExperimentalUnsignedTypes

fun UIntArray.slice(indices: IntRange): List<UInt>
{
    val size = indices.collectionSizeOrDefault(10)
    if (size == 0) return emptyList()
    val list = ArrayList<UInt>(size)
    for (index in indices) { list.add(get(index)) }
    return list
}

* @SinceKotlin("1.3") @ExperimentalUnsignedTypes

fun ULongArray.slice(indices: IntRange): List<ULong>
{
    val size = indices.collectionSizeOrDefault(10)
    if (size == 0) return emptyList()
    val list = ArrayList<ULong>(size)
    for (index in indices) { list.add(get(index)) }
    return list
}

* @SinceKotlin("1.3") @ExperimentalUnsignedTypes

fun UByteArray.slice(indices: IntRange): List<UByte>
{
    val size = indices.collectionSizeOrDefault(10)
    if (size == 0) return emptyList()
    val list = ArrayList<UByte>(size)
    for (index in indices) { list.add(get(index)) }
    return list
}

* @SinceKotlin("1.3") @ExperimentalUnsignedTypes

fun UShortArray.slice(indices: IntRange): List<UShort>
{
    val size = indices.collectionSizeOrDefault(10)
    if (size == 0) return emptyList()
    val list = ArrayList<UShort>(size)
    for (index in indices) { list.add(get(index)) }
    return list
}

* @SinceKotlin("1.3") @ExperimentalUnsignedTypes

fun UIntArray.slice(indices: IntRange): List<UInt>
{
    val size = indices.collectionSizeOrDefault(10)
    if (size == 0) return emptyList()
    val list = ArrayList<UInt>(size)
    for (index in indices) { list.add(get(index)) }
    return list
}

* @SinceKotlin("1.3") @ExperimentalUnsignedTypes

fun ULongArray.slice(indices: IntRange): List<ULong>
{
    val size = indices.collectionSizeOrDefault(10)
    if (size == 0) return emptyList()
    val list = ArrayList<ULong>(size)
    for (index in indices) { list.add(get(index)) }
    return list
}

* @SinceKotlin("1.3") @ExperimentalUnsignedTypes

fun UByteArray.slice(indices: IntRange): List<UByte>
{
    val size = indices.collectionSizeOrDefault(10)
    if (size == 0) return emptyList()
    val list = ArrayList<UByte>(size)
    for (index in indices) { list.add(get(index)) }
    return list
}

* @SinceKotlin("1.3") @ExperimentalUnsignedTypes

fun UShortArray.slice(indices: IntRange): List<UShort>
{
    val size = indices.collectionSizeOrDefault(10)
    if (size == 0) return emptyList()
    val list = ArrayList<UShort>(size)
    for (index in indices) { list.add(get(index)) }
    return list
}

* @SinceKotlin("1.3") @ExperimentalUnsignedTypes

fun UIntArray.slice(indices: IntRange): List<UInt>
{
    val size = indices.collectionSizeOrDefault(10)
    if (size == 0) return emptyList()
    val list = ArrayList<UInt>(size)
    for (index in indices) { list.add(get(index)) }
    return list
}

* @SinceKotlin("1.3") @ExperimentalUnsignedTypes

fun ULongArray.slice(indices: IntRange): List<ULong>
{
    val size = indices.collectionSizeOrDefault(10)
    if (size == 0) return emptyList()
    val list = ArrayList<ULong>(size)
    for (index in indices) { list.add(get(index)) }
    return list
}

* @SinceKotlin("1.3") @ExperimentalUnsignedTypes

fun UByteArray.slice(indices: IntRange): List<UByte>
{
    val size = indices.collectionSizeOrDefault(10)
    if (size == 0) return emptyList()
    val list = ArrayList<UByte>(size)
    for (index in indices) { list.add(get(index)) }
    return list
}

* @SinceKotlin("1.3") @ExperimentalUnsignedTypes

fun UShortArray.slice(indices: IntRange): List<UShort>
{
    val size = indices.collectionSizeOrDefault(10)
    if (size == 0) return emptyList()
    val list = ArrayList<UShort>(size)
    for (index in indices) { list.add(get(index)) }
    return list
}
containing elements of this array at specified [indices].

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.sliceArray(indices: Collection<Int>): UShortArray {
    return UShortArray(storage.sliceArray(indices))
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray.sliceArray(indices: IntRange): UIntArray {
    return UIntArray(storage.sliceArray(indices))
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongArray.sliceArray(indices: IntRange): ULongArray {
    return ULongArray(storage.sliceArray(indices))
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.sliceArray(indices: IntRange): UByteArray {
    return UByteArray(storage.sliceArray(indices))
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.sliceArray(indices: IntRange): UShortArray {
    return UShortArray(storage.sliceArray(indices))
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.sliceArray(indices: IntRange): UByteArray {
    return UByteArray(storage.sliceArray(indices))
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray.take(n: Int): List<UInt> {
    require(n >= 0) {
        "Requested element count $n is less than zero."
    }
    if (n == 0) return emptyList()
    if (n >= size) return toList()
    if (n == 1) return listOf(this[0])
    var count = 0
    val list = ArrayList<UInt>(n)
    for (item in this) {
        list.add(item)
        if (++count == n)
            break
    }
    return list
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongArray.take(n: Int): List<ULong> {
    require(n >= 0) {
        "Requested element count $n is less than zero."
    }
    if (n == 0) return emptyList()
    if (n >= size) return toList()
    if (n == 1) return listOf(this[0])
    var count = 0
    val list = ArrayList<ULong>(n)
    for (item in this) {
        list.add(item)
        if (++count == n)
            break
    }
    return list
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.take(n: Int): List<UByte> {
    require(n >= 0) {
        "Requested element count $n is less than zero."
    }
    if (n == 0) return emptyList()
    if (n >= size) return toList()
    if (n == 1) return listOf(this[0])
    var count = 0
    val list = ArrayList<UByte>(n)
    for (item in this) {
        list.add(item)
        if (++count == n)
            break
    }
    return list
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.take(n: Int): List<UShort> {
    require(n >= 0) {
        "Requested element count $n is less than zero."
    }
    if (n == 0) return emptyList()
    if (n >= size) return toList()
    if (n == 1) return listOf(this[0])
    var count = 0
    val list = ArrayList<UShort>(n)
    for (item in this) {
        list.add(item)
        if (++count == n)
            break
    }
    return list
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray.takeLast(n: Int): List<UInt> {
    require(n >= 0) {
        "Requested element count $n is less than zero."
    }
    if (n == 0) return emptyList()
    if (n >= size) return toList()
    if (n == 1) return listOf(this[size - 1])
    var count = 0
    val list = ArrayList<UInt>(n)
    for (index in size - n until size)
        list.add(this[index])
    return list
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongArray.takeLast(n: Int): List<ULong> {
    require(n >= 0) {
        "Requested element count $n is less than zero."
    }
    if (n == 0) return emptyList()
    if (n >= size) return toList()
    if (n == 1) return listOf(this[size - 1])
    var count = 0
    val list = ArrayList<ULong>(n)
    for (index in size - n until size)
        list.add(this[index])
    return list
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.takeLast(n: Int): List<UByte> {
    require(n >= 0) {
        "Requested element count $n is less than zero."
    }
    if (n == 0) return emptyList()
    if (n >= size) return toList()
    if (n == 1) return listOf(this[size - 1])
    var count = 0
    val list = ArrayList<UByte>(n)
    for (index in size - n until size)
        list.add(this[index])
    return list
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.takeLast(n: Int): List<UShort> {
    require(n >= 0) {
        "Requested element count $n is less than zero."
    }
    if (n == 0) return emptyList()
    if (n >= size) return toList()
    if (n == 1) return listOf(this[size - 1])
    var count = 0
    val list = ArrayList<UShort>(n)
    for (index in size - n until size)
        list.add(this[index])
    return list
}
```

Open Source Used in security_automation identity-guest-cs 1500
{
    require(n >= 0) { \"Requested element count $n is less than zero.\" }

    if (n == 0) return emptyList()
    val size = size
    if (n >= size) return toList()
    if (n == 1) return listOf(this[size - 1])
    val list = ArrayList<ULong>(n)
    for (index in size - n until size)
        list.add(this[index])
    return list
}

/**
 * Returns a list containing last [n] elements.
 * 
 * @throws IllegalArgumentException if [n] is negative.
 * 
 * @sample samples.collections.Collections.Transformations.take
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.takeLast(n: Int): List<UByte> {
    require(n >= 0) { \"Requested element count $n is less than zero.\" }
    if (n == 0) return emptyList()
    val size = size
    if (n >= size) return toList()
    if (n == 1) return listOf(this[size - 1])
    val list = ArrayList<UByte>(n)
    for (index in size - n until size)
        list.add(this[index])
    return list
}

/**
 * Returns a list containing last [n] elements satisfying the given [predicate].
 * 
 * @sample samples.collections.Collections.Transformations.take
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.takeLastWhile(predicate: (UInt) -> Boolean): List<UInt> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return drop(index + 1)
        }
    }
    return toList()
}

/**
 * Returns a list containing first elements satisfying the given [predicate].
 * 
 * @sample samples.collections.Collections.Transformations.take
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.takeWhile(predicate: (UInt) -> Boolean): List<UInt> {
    val list = ArrayList<UInt>()
    for (item in this) {
        if (!predicate(item))
            break
        list.add(item)
    }
    return list
}

/**
 * Returns a list containing first elements satisfying the given [predicate].
 * 
 * @sample samples.collections.Collections.Transformations.take
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.takeWhile(predicate: (ULong) -> Boolean): List<ULong> {
    val list = ArrayList<ULong>()
    for (item in this) {
        if (!predicate(item))
            break
        list.add(item)
    }
    return list
}

/**
 * Returns a list containing first elements satisfying the given [predicate].
 * 
 * @sample samples.collections.Collections.Transformations.take
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.takeLastWhile(predicate: (UByte) -> Boolean): List<UByte> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return drop(index + 1)
        }
    }
    return toList()
}

/**
 * Returns a list containing first elements satisfying the given [predicate].
 * 
 * @sample samples.collections.Collections.Transformations.take
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.takeLastWhile(predicate: (UShort) -> Boolean): List<UShort> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return drop(index + 1)
        }
    }
    return toList()
}
UByteArray.takeWhile(predicate: (UByte) -> Boolean): List<UByte> {
    val list = ArrayList<UByte>()
    for (item in this) {
        if (!predicate(item))
            break
        list.add(item)
    }
    return list
}

/**
 * Returns a list containing first elements satisfying the given [predicate].
 * @sample
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.takeWhile(predicate: (UByte) -> Boolean): List<UByte> {
    val list = ArrayList<UByte>()
    for (item in this) {
        if (!predicate(item))
            break
        list.add(item)
    }
    return list
}

/**
 * Returns a list containing first elements satisfying the given [predicate].
 * @sample
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.takeWhile(predicate: (UShort) -> Boolean): List<UShort> {
    val list = ArrayList<UShort>()
    for (item in this) {
        if (!predicate(item))
            break
        list.add(item)
    }
    return list
}

/**
 * Reverses elements in the array in-place.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.reverse(): Unit {
    storage.reverse()
}

/**
 * Reverses elements in the array in-place.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.reverse(): Unit {
    storage.reverse()
}

/**
 * Reverses elements in the array in-place.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.reverse(): Unit {
    storage.reverse()
}

/**
 * Reverses elements in the array in-place.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.reverse(): Unit {
    storage.reverse()
}

/**
 * Reverses elements of the array in the specified range in-place.
 * @param fromIndex the start of the range (inclusive) to reverse.
 * @param toIndex the end of the range (exclusive) to reverse.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.reverse(fromIndex: Int, toIndex: Int): Unit {
    storage.reverse(fromIndex, toIndex)
}

/**
 * Reverses elements of the array in the specified range in-place.
 * @param fromIndex the start of the range (inclusive) to reverse.
 * @param toIndex the end of the range (exclusive) to reverse.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.reverse(fromIndex: Int, toIndex: Int): Unit {
    storage.reverse(fromIndex, toIndex)
}

/**
 * Reverses elements of the array in the specified range in-place.
 * @param fromIndex the start of the range (inclusive) to reverse.
 * @param toIndex the end of the range (exclusive) to reverse.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.reverse(fromIndex: Int, toIndex: Int): Unit {
    storage.reverse(fromIndex, toIndex)
}

/**
 * Reverses elements of the array in the specified range in-place.
 * @param fromIndex the start of the range (inclusive) to reverse.
 * @param toIndex the end of the range (exclusive) to reverse.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.reverse(fromIndex: Int, toIndex: Int): Unit {
    storage.reverse(fromIndex, toIndex)
}

/**
 * Returns a list with elements in reversed order.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray.reversed(): List<UInt> {
    if (isEmpty()) return emptyList()
    val list = toMutableList()
    list.reverse()
    return list
}

/**
 * Returns a list with elements in reversed order.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongArray.reversed(): List<ULong> {
    if (isEmpty()) return emptyList()
    val list = toMutableList()
    list.reverse()
    return list
}

/**
 * Returns a list with elements in reversed order.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.reversed(): List<UByte> {
    if (isEmpty()) return emptyList()
    val list = toMutableList()
    list.reverse()
    return list
}

/**
 * Returns a list with elements in reversed order.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.reversed(): List<UShort> {
    if (isEmpty()) return emptyList()
    val list = toMutableList()
    list.reverse()
    return list
}
Open Source Used In security_automation identity-guest-cs

```kotlin
if (isEmpty()) return emptyList()
val list = toMutableList()
list.reverse()
return list

/**
 * Returns an array with elements of this array in reversed order.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.reversedArray(): UIntArray {
    return UIntArray(storage.reversedArray())
}

/**
 * Returns an array with elements of this array in reversed order.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.reversedArray(): ULongArray {
    return ULongArray(storage.reversedArray())
}

/**
 * Returns an array with elements of this array in reversed order.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.reversedArray(): UByteArray {
    return UByteArray(storage.reversedArray())
}

/**
 * Returns an array with elements of this array in reversed order.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.reversedArray(): UShortArray {
    return UShortArray(storage.reversedArray())
}

/**
 * Randomly shuffles elements in this array in-place.
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
public fun UIntArray.shuffle(): Unit {
    shuffle(Random)
}

/**
 * Randomly shuffles elements in this array in-place.
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
public fun ULongArray.shuffle(): Unit {
    shuffle(Random)
}

/**
 * Randomly shuffles elements in this array in-place.
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
public fun UByteArray.shuffle(): Unit {
    shuffle(Random)
}

/**
 * Randomly shuffles elements in this array in-place.
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
public fun UShortArray.shuffle(): Unit {
    shuffle(Random)
}

/**
 * Sorts elements in the array in-place descending according to their natural sort order.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray.sortDescending(): Unit {
    if (size > 1) {
        sort() reverse() 
    }
}

/**
 * Sorts elements in the array in-place descending according to their natural sort order.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongArray.sortDescending(): Unit {
    if (size > 1) {
        sort() reverse() 
    }
}
```

---

**Open Source Used In security_automation identity-guest-cs**

1503
UBYTEARRAY.sortDescending(): Unit |\n  if (size > 1) { |\n    sort() |\n    reverse() |\n  } |\n  * Sorts elements in the array in-place descending according to their natural sort order.

*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes@public fun USHORTARRAY.sortDescending(): Unit |\n  if (size > 1) { |\n    sort() |\n    reverse() |\n  } |\n  * Returns a list of all elements sorted according to their natural sort order.

*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes@public fun UINTARRAY.sorted(): List<UINT> |\n  return copyOf().apply { |\n    sort() |\n  } |\n  .asList() |\n  * Returns a list of all elements sorted according to their natural sort order.

*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes@public fun ULONGARRAY.sorated(): List<ULONG> |\n  return copyOf().apply { |\n    sort() |\n  } |\n  .asList() |\n  * Returns a list of all elements sorted according to their natural sort order.

*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes@public fun UBYTEARRAY.sorated(): List<UBYTE> |\n  return copyOf().apply { |\n    sort() |\n  } |\n  .asList() |\n  * Returns a list of all elements sorted according to their natural sort order.

*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes@public fun USHORTARRAY.sorated(): List<USHORT> |\n  return copyOf().apply { |\n    sort() |\n  } |\n  .asList() |\n  * Returns a list of all elements sorted according to their natural sort order.

*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes@public fun UINTARRAY.sortedArray(): UINTARRAY |\n  if (isEmpty()) return this |\n  return this.copyOf().apply { |\n    sort() |\n  } |\n  * Returns an array with all elements of this array sorted according to their natural sort order.

*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes@public fun ULONGARRAY.sortedArray(): ULONGARRAY |\n  if (isEmpty()) return this |\n  return this.copyOf().apply { |\n    sort() |\n  } |\n  * Returns an array with all elements of this array sorted according to their natural sort order.

*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes@public fun UBYTEARRAY.sortedArray(): UBYTEARRAY |\n  if (isEmpty()) return this |\n  return this.copyOf().apply { |\n    sort() |\n  } |\n  * Returns an array with all elements of this array sorted according to their natural sort order.

*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes@public fun USHORTARRAY.sortedArray(): USHORTARRAY |\n  if (isEmpty()) return this |\n  return this.copyOf().apply { |\n    sort() |\n  } |\n  * Returns an array with all elements of this array sorted according to their natural sort order.

*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes@public fun UINTARRAY.sortedArrayDescending(): UINTARRAY |\n  if (isEmpty()) return this |\n  return this.copyOf().apply { |\n    sortDescending() |\n  } |\n  * Returns an array with all elements of this array sorted descending according to their natural sort order.

*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes@public fun ULONGARRAY.sortedArrayDescending(): ULONGARRAY |\n  if (isEmpty()) return this |\n  return this.copyOf().apply { |\n    sortDescending() |\n  } |\n  * Returns a list of all elements sorted descending according to their natural sort order.

*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes@public fun USHORTARRAY.sortedArrayDescending(): USHORTARRAY |\n  if (isEmpty()) return this |\n  return this.copyOf().apply { |\n    sortDescending() |\n  } |\n  * Returns a list of all elements sorted descending according to their natural sort order.

*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes@public fun UBYTEARRAY.sortedArrayDescending(): UBYTEARRAY |\n  if (isEmpty()) return this |\n  return this.copyOf().apply { |\n    sortDescending() |\n  } |\n  * Returns a list of all elements sorted descending according to their natural sort order.

*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes@public fun UINTARRAY.sortedDescending(): List<UINT> |\n  return copyOf().apply { |\n    sort() |\n  } |\n  .reversed() |\n  * Returns a list of all elements sorted descending according to their natural sort order.

*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes@public fun ULONGARRAY.sortedDescending(): List<ULONG> |\n  return copyOf().apply { |\n    sort() |\n  } |\n  .reversed() |\n  * Returns a list of all elements sorted descending according to their natural sort order.

*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes@public fun UBYTEARRAY.sortedDescending(): List<UBYTE> |\n  return copyOf().apply { |\n    sort() |\n  } |\n  .reversed() |\n  * Returns a list of all elements sorted descending according to their natural sort order.
descending according to their natural sort order. The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting. The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.sortedDescending(): List<UShort> {
    return copyOf().apply { sort() }.reversed()
}
```

```kotlin
/**
 * Returns an array of type [ByteArray], which is a view of this array where each element is a signed reinterpretation of the corresponding element of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public expect fun UByteArray.asList(): List<UByte>
```

```kotlin
/**
 * Returns an array of type [UIntArray], which is a view of this array where each element is a signed reinterpretation of the corresponding element of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public expect fun UIntArray.asList(): List<UInt>
```

```kotlin
/**
 * Returns an array of type [ULongArray], which is a view of this array where each element is a signed reinterpretation of the corresponding element of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public expect fun ULongArray.asList(): List<ULong>
```

```kotlin
/**
 * Returns an array of type [UByteArray], which is a view of this array where each element is an unsigned reinterpretation of the corresponding element of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline fun ByteArray.asUByteArray(): UByteArray {
    return UByteArray(this)
}
```

```kotlin
/**
 * Returns an array of type [UIntArray], which is a view of this array where each element is an unsigned reinterpretation of the corresponding element of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun IntArray.asUIntArray(): UIntArray {
    return UIntArray(this)
}
```

```kotlin
/**
 * Returns an array of type [ULongArray], which is a view of this array where each element is an unsigned reinterpretation of the corresponding element of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun LongArray.asULongArray(): ULongArray {
    return ULongArray(this)
}
```

The sort order is used to sort the elements in the array. The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting. The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.
Boolean { return this.contentEquals(other)}

*structurally* equal to one another, 
* i.e. contain the same number of the same elements in the same order.

@SinceKotlin("1.3")
@Deprecation
public fun UByteArray.contentEquals(): Boolean {
    return this.contentEquals() }
is

```kotlin
open fun UShortArray?.contentHashCode(): Int {
    return this?.storage.contentHashCode()
}

@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
public fun UShortArray?.contentHashCode(): Int {
    return this?.storage.contentHashCode()
}

/**
 * Returns a string representation of the contents of the specified array as if it is 
 * [List].
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
public fun UIntArray?.contentHashCode(): Int {
    return this?.storage.contentHashCode()
}

/**
 * Returns a string representation of the contents of the specified array as if it is 
 * [List].
 * 
 * @sample samples.collections.Arrays.ContentOperations.contentToString
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray?.contentHashCode(): Int {
    return this?.storage.contentHashCode()
}

/**
 * Returns a string representation of the contents of the specified array as if it is 
 * [List].
 * 
 * @sample samples.collections.Arrays.ContentOperations.contentToString
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongArray?.contentHashCode(): Int {
    return this?.storage.contentHashCode()
}

/**
 * Returns a string representation of the contents of the specified array as if it is 
 * [List].
 * 
 * @sample samples.collections.Arrays.ContentOperations.contentToString
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray?.contentHashCode(): Int {
    return this?.storage.contentHashCode()
}

/**
 * Returns a string representation of the contents of the specified array as if it is 
 * [List].
 * 
 * @sample samples.collections.Arrays.ContentOperations.contentToString
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray?.contentHashCode(): Int {
    return this?.storage.contentHashCode()
}

/**
 * Copies this array or its subrange into the 
 * [destination] array and returns that array.
 * 
 * It's allowed to pass the same array in the [destination] and even
 * specify the subrange so that it overlaps with the destination range.
 * @param destination the array to copy to
 * @param destinationOffset the position in the [destination] array to copy to, 0 by default
 * @param startIndex the beginning (inclusive) of the subrange to copy, size of this array by default.
 * @param endIndex the end (exclusive) of the subrange to copy, zero by default.
 * 
 * @throws IndexOutOfBoundsException or 
 * [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex
 * > endIndex`.
 * @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array
 * starting at the specified [destinationOffset].
 * @param destinationOffset, or when that index is out of the [destination] array indices range.
 * @return the [destination] array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
```
```kotlin
UIntArray {
    storage.copyInto(destination.storage, destinationOffset, startIndex, endIndex)
    return
}

/**
 * Copies this array or its subrange into the [destination] array and returns that array.
 * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with
 * the destination range.
 * @param destination the array to copy to.
 * @param destinationOffset the position in the [destination] array to copy to, 0 by default.
 * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.
 * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.
 * @return the [destination] array.
 * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of
 * range of this array indices or when `startIndex > endIndex`.
 * @throws IndexOutOfRangeBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset],
 * or when that index is out of the [destination] array indices range.
 * @return the [destination] array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.copyInto(destination: UIntArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): UIntArray {
    storage.copyInto(destination.storage, destinationOffset, startIndex, endIndex)
    return
}

/**
 * Copies this array or its subrange into the [destination] array and returns that array.
 * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with
 * the destination range.
 * @param destination the array to copy to.
 * @param destinationOffset the position in the [destination] array to copy to, 0 by default.
 * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.
 * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.
 * @return the [destination] array.
 * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of
 * range of this array indices or when `startIndex > endIndex`.
 * @throws IndexOutOfRangeBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset],
 * or when that index is out of the [destination] array indices range.
 * @return the [destination] array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.copyInto(destination: ULongArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): ULongArray {
    storage.copyInto(destination.storage, destinationOffset, startIndex, endIndex)
    return
}

/**
 * Copies this array or its subrange into the [destination] array and returns that array.
 * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with
 * the destination range.
 * @param destination the array to copy to.
 * @param destinationOffset the position in the [destination] array to copy to, 0 by default.
 * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.
 * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.
 * @return the [destination] array.
 * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of
 * range of this array indices or when `startIndex > endIndex`.
 * @throws IndexOutOfRangeBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset],
 * or when that index is out of the [destination] array indices range.
 * @sample samples.collections.Arrays.CopyOfOperations.copyOf
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.copyInto(destination: UByteArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): UByteArray {
    storage.copyInto(destination.storage, destinationOffset, startIndex, endIndex)
    return
}

/**
 * Copies this array or its subrange into the [destination] array and returns that array.
 * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with
 * the destination range.
 * @param destination the array to copy to.
 * @param destinationOffset the position in the [destination] array to copy to, 0 by default.
 * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.
 * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.
 * @return the [destination] array.
 * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of
 * range of this array indices or when `startIndex > endIndex`.
 * @throws IndexOutOfRangeBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset],
 * or when that index is out of the [destination] array indices range.
 * @sample samples.collections.Arrays.CopyOfOperations.copyOf
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.copyInto(destination: UShortArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): UShortArray {
    storage.copyInto(destination.storage, destinationOffset, startIndex, endIndex)
    return
}

/**
 * Returns new array which is a copy of the original array.
 * @sample samples.collections.Arrays.CopyOfOperations.copyOf
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.copyOf(): UIntArray {
    return UIntArray(storage.copyOf())
}

/**
 * Returns new array which is a copy of the original array.
 * @sample samples.collections.Arrays.CopyOfOperations.copyOf
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.copyOf(): ULongArray {
    return ULongArray(storage.copyOf())
}

/**
 * Returns new array which is a copy of the original array.
 * @sample samples.collections.Arrays.CopyOfOperations.copyOf
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.copyOf(): UByteArray {
    return UByteArray(storage.copyOf())
}

/**
 * Returns new array which is a copy of the original array.
 * @sample samples.collections.Arrays.CopyOfOperations.copyOf
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.copyOf(): UShortArray {
    return UShortArray(storage.copyOf())
}
```

Open Source Used In security_automation identity-guest-cs 1508
*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly
public inline fun UShortArray.copyOf(): UShortArray {
    return UShortArray(storage.copyOf())
}

/**
 * Returns new array which is a copy of the original array, resized to the given [newSize].
 * The copy is either truncated or padded at the end with zero values if necessary.
 * - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].
 * - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with zero values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.copyOf(newSize: Int): UIntArray {
    return UIntArray(storage.copyOf(newSize))
}

/**
 * Returns new array which is a copy of the original array, resized to the given [newSize].
 * The copy is either truncated or padded at the end with zero values if necessary.
 * - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].
 * - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with zero values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.copyOf(newSize: Int): ULongArray {
    return ULongArray(storage.copyOf(newSize))
}

/**
 * Returns new array which is a copy of the original array, resized to the given [newSize].
 * The copy is either truncated or padded at the end with zero values if necessary.
 * - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].
 * - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with zero values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.copyOf(newSize: Int): UByteArray {
    return UByteArray(storage.copyOf(newSize))
}

/**
 * Returns a new array which is a copy of the specified range of the original array.
 * @param fromIndex the start of the range (inclusive) to copy.
 * @param toIndex the end of the range (exclusive) to copy.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.copyOfRange(fromIndex: Int, toIndex: Int): UIntArray {
    return UIntArray(storage.copyOfRange(fromIndex, toIndex))
}

/**
 * Returns a new array which is a copy of the specified range of the original array.
 * @param fromIndex the start of the range (inclusive) to copy.
 * @param toIndex the end of the range (exclusive) to copy.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.copyOfRange(fromIndex: Int, toIndex: Int): ULongArray {
    return ULongArray(storage.copyOfRange(fromIndex, toIndex))
}

/**
 * Returns a new array which is a copy of the specified range of the original array.
 * @param fromIndex the start of the range (inclusive) to copy.
 * @param toIndex the end of the range (exclusive) to copy.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.copyOfRange(fromIndex: Int, toIndex: Int): UByteArray {
    return UByteArray(storage.copyOfRange(fromIndex, toIndex))
}
@param toIndex the end of the range (exclusive) to copy.
@throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
@throws IllegalArgumentException if [fromIndex] is greater than [toIndex].

@SinceKotlin("1.3") fun @ExperimentalUnsignedTypes @kotlin.internal:inline fun UShortArray.copyOfRange(fromIndex: Int, toIndex: Int): UShortArray

@SinceKotlin("1.3") fun @ExperimentalUnsignedTypes public fun UIntArray.fill(element: UInt, fromIndex: Int = 0, toIndex: Int = size): Unit

@SinceKotlin("1.3") fun @ExperimentalUnsignedTypes public fun ULongArray.fill(element: ULong, fromIndex: Int = 0, toIndex: Int = size): Unit

@SinceKotlin("1.3") fun @ExperimentalUnsignedTypes public fun UByteArray.fill(element: UByte, fromIndex: Int = 0, toIndex: Int = size): Unit

@SinceKotlin("1.3") fun @ExperimentalUnsignedTypes public fun UShortArray.fill(element: UShort, fromIndex: Int = 0, toIndex: Int = size): Unit

@SinceKotlin("1.3") fun @ExperimentalUnsignedTypes inline fun UIntArray.indices: IntRange
    get() = storage.indices

@SinceKotlin("1.3") fun @ExperimentalUnsignedTypes inline fun ULongArray.indices: IntRange
    get() = storage.indices

@SinceKotlin("1.3") fun @ExperimentalUnsignedTypes inline fun UByteArray.indices: IntRange
    get() = storage.indices

@SinceKotlin("1.3") fun @ExperimentalUnsignedTypes inline fun UShortArray.indices: IntRange
    get() = storage.indices

@SinceKotlin("1.3") fun @ExperimentalUnsignedTypes inline fun UIntArray.lastIndex: Int
    get() = storage.lastIndex

@SinceKotlin("1.3") fun @ExperimentalUnsignedTypes inline fun ULongArray.lastIndex: Int
    get() = storage.lastIndex

@SinceKotlin("1.3") fun @ExperimentalUnsignedTypes inline fun UByteArray.lastIndex: Int
    get() = storage.lastIndex

@SinceKotlin("1.3") fun @ExperimentalUnsignedTypes inline fun UShortArray.lastIndex: Int
    get() = storage.lastIndex

@SinceKotlin("1.3") fun @ExperimentalUnsignedTypes inline fun UIntArray.copyOfRange(fromIndex: Int, toIndex: Int): UIntArray

@SinceKotlin("1.3") fun @ExperimentalUnsignedTypes inline fun ULongArray.copyOfRange(fromIndex: Int, toIndex: Int): ULongArray

@SinceKotlin("1.3") fun @ExperimentalUnsignedTypes inline fun UByteArray.copyOfRange(fromIndex: Int, toIndex: Int): UByteArray

@SinceKotlin("1.3") fun @ExperimentalUnsignedTypes inline fun UShortArray.copyOfRange(fromIndex: Int, toIndex: Int): UShortArray

operator fun UIntArray.plus(element: UInt): UIntArray {
    return UIntArray(storage + element.toInt())
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public operator fun UIntArray.plus(elements: Collection<UInt>): UIntArray {
    var index = size
    val result = storage.copyOf(size + elements.size)
    for (element in elements) result[index++] = element.toInt()
    return UIntArray(result)
}

/**
 * Sorts the array in-place.
 * 
 * @sample samples.collections.Arrays.Sorting.sortArray
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray.sort(): Unit {
    if (size > 1) sortArray(this, 0, size)
}

operator fun ULongArray.plus(element: ULong): ULongArray {
    return ULongArray(storage + element.toLong())
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public operator fun ULongArray.plus(elements: Collection<ULong>): ULongArray {
    var index = size
    val result = storage.copyOf(size + elements.size)
    for (element in elements) result[index++] = element.toLong()
    return ULongArray(result)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public operator fun UByteArray.plus(element: UByte): UByteArray {
    return UByteArray(storage + element.toByte())
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public operator fun UByteArray.plus(elements: Collection<UByte>): UByteArray {
    var index = size
    val result = storage.copyOf(size + elements.size)
    for (element in elements) result[index++] = element.toByte()
    return UByteArray(result)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public operator fun UShortArray.plus(element: UShort): UShortArray {
    return UShortArray(storage + element.toShort())
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public operator fun UShortArray.plus(elements: Collection<UShort>): UShortArray {
    var index = size
    val result = storage.copyOf(size + elements.size)
    for (element in elements) result[index++] = element.toShort()
    return UShortArray(result)
}
fun UByteArray.sort(): Unit {
    if (size > 1)
        sortArray(this, 0, size)
}

/**
 * Sorts the array in-place.
 * @param fromIndex the start of the range (inclusive) to sort, 0 by default.
 * @param toIndex the end of the range (exclusive) to sort, size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 * @sample samples.collections.Arrays.Sorting.sortArray
*/
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.sort(): Unit {
    if (size > 1)
        sortArray(this, 0, size)
}

/**
 * Sorts the array in-place.
 * @param fromIndex the start of the range (inclusive) to sort, 0 by default.
 * @param toIndex the end of the range (exclusive) to sort, size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 * @sample samples.collections.Arrays.Sorting.sortArray
*/
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray.sort(): Unit {
    if (size > 1)
        sortArray(this, 0, size)
}

/**
 * Sorts a range in the array in-place.
 * @param fromIndex the start of the range (inclusive) to sort, 0 by default.
 * @param toIndex the end of the range (exclusive) to sort, size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 * @sample samples.collections.Arrays.Sorting.sortRangeOfArray
*/
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
public fun ULongArray.sort(fromIndex: Int = 0, toIndex: Int = size): Unit {
    sortArray(this, fromIndex, toIndex)
}

/**
 * Sorts a range in the array in-place.
 * @param fromIndex the start of the range (inclusive) to sort, 0 by default.
 * @param toIndex the end of the range (exclusive) to sort, size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 * @sample samples.collections.Arrays.Sorting.sortRangeOfArray
*/
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
public fun UByteArray.sort(fromIndex: Int = 0, toIndex: Int = size): Unit {
    sortArray(this, fromIndex, toIndex)
}

/**
 * Sorts a range in the array in-place.
 * @param fromIndex the start of the range (inclusive) to sort, 0 by default.
 * @param toIndex the end of the range (exclusive) to sort, size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 * @sample samples.collections.Arrays.Sorting.sortRangeOfArray
*/
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
public fun UShortArray.sort(fromIndex: Int = 0, toIndex: Int = size): Unit {
    sortArray(this, fromIndex, toIndex)
}

/**
 * Sorts elements of the array in the specified range in-place.
 * The elements are sorted descending according to their natural sort order.
 * @param fromIndex the start of the range (inclusive) to sort.
 * @param toIndex the end of the range (exclusive) to sort.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 * @sample samples.collections.Arrays.Sorting.sortDescending
*/
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
public fun UIntArray.sortDescending(fromIndex: Int, toIndex: Int): Unit {
    sort(fromIndex, toIndex)
    reverse(fromIndex, toIndex)
}

/**
 * Sorts elements of the array in the specified range in-place.
 * The elements are sorted descending according to their natural sort order.
 * @param fromIndex the start of the range (inclusive) to sort.
 * @param toIndex the end of the range (exclusive) to sort.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 * @sample samples.collections.Arrays.SortDescending
*/
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
public fun ULongArray.sortDescending(fromIndex: Int, toIndex: Int): Unit {
    sort(fromIndex, toIndex)
    reverse(fromIndex, toIndex)
}
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
public fun UByteArray.sortDescending(fromIndex: Int, toIndex: Int): Unit {
    sort(fromIndex, toIndex)
    reverse(fromIndex, toIndex)
}

/**
 * Sorts elements of the array in the specified range in-place.
 * The elements are sorted descending according to their natural sort order.
 * @param fromIndex the start of the range (inclusive) to sort.
 * @param toIndex the end of the range (exclusive) to sort.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegallegalArgumentException if [fromIndex] is greater than [toIndex].
 * Returns an array of type [ByteArray], which is a copy of this array where each element is a signed reinterpretation
 * of the corresponding element of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline fun UByteArray.toByteArray(): ByteArray {
    return storage.copyOf()
}

/**
 * Returns an array of type [ByteArray], which is a copy of this array where each element is a signed reinterpretation
 * of the corresponding element of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.toByteArray(): ByteArray {
    return storage.copyOf()
}

/**
 * Returns an array of type [ByteArray], which is a copy of this array where each element is a signed reinterpretation
 * of the corresponding element of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.toByteArray(): ByteArray {
    return storage.copyOf()
}

/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.toTypedArray(): Array<UByte> {
    return Array(size) { index -> this[index] }
}

/**
 * Returns an array of type [ByteArray], which is a copy of this array where each element is a signed reinterpretation
 * of the corresponding element of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.toByteArray(): ByteArray {
    return storage.copyOf()
}

/**
 * Returns an array of type [ByteArray], which is a copy of this array where each element is a signed reinterpretation
 * of the corresponding element of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.toByteArray(): ByteArray {
    return storage.copyOf()
}

/**
 * Returns an array of type [ByteArray], which is a copy of this array where each element is a signed reinterpretation
 * of the corresponding element of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.toByteArray(): ByteArray {
    return storage.copyOf()
}

/**
 * Returns an array of type [ByteArray], which is a copy of this array where each element is a signed reinterpretation
 * of the corresponding element of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.toByteArray(): ByteArray {
    return storage.copyOf()
}

/**
 * Returns an array of type [ByteArray], which is a copy of this array where each element is a signed reinterpretation
 * of the corresponding element of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.toByteArray(): ByteArray {
    return storage.copyOf()
}

/**
 * Returns an array of type [ByteArray], which is a copy of this array where each element is a signed reinterpretation
 * of the corresponding element of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.toByteArray(): ByteArray {
    return storage.copyOf()
}

/**
 * Returns an array of type [ByteArray], which is a copy of this array where each element is a signed reinterpretation
 * of the corresponding element of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.toByteArray(): ByteArray {
    return storage.copyOf()
}

/**
 * Returns an array of type [ByteArray], which is a copy of this array where each element is a signed reinterpretation
 * of the corresponding element of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.toByteArray(): ByteArray {
    return storage.copyOf()
}

/**
 * Returns an array of type [ByteArray], which is a copy of this array where each element is a signed reinterpretation
 * of the corresponding element of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.toByteArray(): ByteArray {
    return storage.copyOf()
}

/**
 * Returns an array of type [ByteArray], which is a copy of this array where each element is a signed reinterpretation
 * of the corresponding element of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.toByteArray(): ByteArray {
    return storage.copyOf()
}

/**
 * Returns an array of type [ByteArray], which is a copy of this array where each element is a signed reinterpretation
 * of the corresponding element of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.toByteArray(): ByteArray {
    return storage.copyOf()
}

/**
 * Returns an array of type [ByteArray], which is a copy of this array where each element is a signed reinterpretation
 * of the corresponding element of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.toByteArray(): ByteArray {
    return storage.copyOf()
}

/**
 * Returns an array of type [ByteArray], which is a copy of this array where each element is a signed reinterpretation
 * of the corresponding element of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.toByteArray(): ByteArray {
    return storage.copyOf()
}

/**
 * Returns an array of type [ByteArray], which is a copy of this array where each element is a signed reinterpretation
 * of the corresponding element of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.toByteArray(): ByteArray {
    return storage.copyOf()
}

/**
 * Returns an array of type [ByteArray], which is a copy of this array where each element is a signed reinterpretation
 * of the corresponding element of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.toByteArray(): ByteArray {
    return storage.copyOf()
}

/**
 * Returns an array of type [ByteArray], which is a copy of this array where each element is a signed reinterpretation
 * of the corresponding element of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.toByteArray(): ByteArray {
    return storage.copyOf()
}

/**
 * Returns an array of type [ByteArray], which is a copy of this array where each element is a signed reinterpretation
 * of the corresponding element of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.toByteArray(): ByteArray {
    return storage.copyOf()
}
ULongArray {n return ULongArray(size) { index -> this[index] }n}n/**\n * Returns an array of type [ULongArray], which is a copy of this array where each element is an unsigned reinterpretation
 * of the corresponding element of this array.n
 */

@SinceKotlin("1.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly
public inline fun LongArray.toULongArray(): ULongArray {
    return ULongArray(this.copyOf())
}

/**
 * Returns an array of type
 * [ULongArray], which is a copy of this array where each element is an unsigned reinterpretation
 * of the corresponding element of this array.n
 */

@SinceKotlin("1.3")@ExperimentalUnsignedTypes@public fun Array<out ULong>.toULongArray(): ULongArray {
    return ULongArray(size) { index -> this[index] }
}

/**
 * Returns an array of type
 * [UShortArray], which is a copy of this array where each element is an unsigned reinterpretation
 * of the corresponding element of this array.n
 */

@SinceKotlin("1.3")@ExperimentalUnsignedTypes@public fun ShortArray.toUShortArray(): UShortArray {
    return UShortArray(size) { index -> this[index] }
}

/**
 * Returns an array of type
 * [UShortArray], which is a copy of this array where each element is an unsigned reinterpretation
 * of the corresponding element of this array.n
 */

@SinceKotlin("1.3")@ExperimentalUnsignedTypes@public inline fun UIntArray.associateWith(valueSelector: (UInt) -> V): Map<UInt, V> {
    val result = LinkedHashMap<UInt, V>(mapCapacity(size).coerceAtLeast(16))
    return associateWithTo(result, valueSelector)
}

/**
 * Populates and returns the [destination] mutable map with key-value pairs for each element of the given array,
 * where key is the element itself and value is provided by the [valueSelector] function applied to each element.n
 * If any two elements are equal, the last one gets added to the map.n
 * The returned map preserves the entry iteration order of the original array.n
 */

@SinceKotlin("1.4")@ExperimentalUnsignedTypes@public inline fun <V, M : MutableMap<in UInt, in V>> UIntArray.associateWithTo(destination: M, valueSelector: (UInt) -> V): M {
    for (element in this) {
        destination.put(element, valueSelector(element))
    }
    return destination
}

/**
 * Populates and returns the [destination] mutable map with key-value pairs for each element of the given array,
 * where key is the element itself and value is provided by the [valueSelector] function applied to each element.n
 * If any two elements are equal, the last one gets added to the map.n
 * The returned map preserves the entry iteration order of the original array.
 */

@SinceKotlin("1.4")@ExperimentalUnsignedTypes@public inline fun <V, M : MutableMap<in UShort, in V>> UShortArray.associateWithTo(destination: M, valueSelector: (UShort) -> V): M {
    for (element in this) {
        destination.put(element, valueSelector(element))
    }
    return destination
}
where key is the element itself and value is provided by the [valueSelector] function applied to that key.

If any two elements are equal, the last one overwrites the former value in the map.

@sample samples.collections.Collections.Transformations.associateWithTo

@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <V, M : MutableMap<in ULong, in V>> ULongArray.associateWithTo(destination: M, valueSelector: (ULong) -> V): M {
    for (element in this) {
        destination.put(element, valueSelector(element))
    }
    return destination
}

@sample samples.collections.Collections.Transformations.associateWithTo

@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <V, M : MutableMap<in UByte, in V>> UByteArray.associateWithTo(destination: M, valueSelector: (UByte) -> V): M {
    for (element in this) {
        destination.put(element, valueSelector(element))
    }
    return destination
}

@sample samples.collections.Collections.Transformations.associateWithTo

@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <V, M : MutableMap<in UShort, in V>> UShortArray.associateWithTo(destination: M, valueSelector: (UShort) -> V): M {
    for (element in this) {
        destination.put(element, valueSelector(element))
    }
    return destination
}

@sample samples.collections.Collections.Transformations.associateWithTo

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UIntArray.flatMap(transform: (UInt) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}

@sample samples.collections.Collections.Transformations.flatMap

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> ULongArray.flatMap(transform: (ULong) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}

@sample samples.collections.Collections.Transformations.flatMap

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UByteArray.flatMap(transform: (UByte) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}

@sample samples.collections.Collections.Transformations.flatMap

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UShortArray.flatMap(transform: (UShort) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}

@sample samples.collections.Collections.Transformations.flatMap

@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@kotlin.experimental.ExperimentalTypeInference
public inline fun <R> UIntArray.flatMapIndexed(transform: (index: Int, UInt) -> Iterable<R>): List<R> {
    return flatMapIndexedTo(ArrayList<R>(), transform)
}

@sample samples.collections.Collections.Transformations.flatMapIndexed

@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@kotlin.experimental.ExperimentalTypeInference
public inline fun <R> ULongArray.flatMapIndexed(transform: (index: Int, ULong) -> Iterable<R>): List<R> {
    return flatMapIndexedTo(ArrayList<R>(), transform)
}

@sample samples.collections.Collections.Transformations.flatMapIndexed

@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@kotlin.experimental.ExperimentalTypeInference
public inline fun <R> UByteArray.flatMapIndexed(transform: (index: Int, UByte) -> Iterable<R>): List<R> {
    return flatMapIndexedTo(ArrayList<R>(), transform)
}

@sample samples.collections.Collections.Transformations.flatMapIndexed

@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@kotlin.experimental.ExperimentalTypeInference
public inline fun <R> UShortArray.flatMapIndexed(transform: (index: Int, UShort) -> Iterable<R>): List<R> {
    return flatMapIndexedTo(ArrayList<R>(), transform)
}

@sample samples.collections.Collections.Transformations.flatMapIndexed

@openSourceIn("security_automation
identity-guest-cs") 1515
@sample samples.collections.Collections.Transformations.flatMapIndexed

*\@SinceKotlin("1.4\")*\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\public inline fun \<R>
ULongArray.flatMapIndexed(transform: (index: Int, ULong) -> Iterable\<R>)(transform): List\<R> { \
    var list = transform(index++, element)
    return List\<R>()
}

*\@SinceKotlin("1.4\")*\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\public inline fun \<R>
UByteArray.flatMapIndexed(transform: (index: Int, UByte) -> Iterable\<R>)(transform): List\<R> { \
    var list = transform(index++, element)
    return List\<R>()
}

*\@SinceKotlin("1.4\")*\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\public inline fun \<R>
UShortArray.flatMapIndexed(transform: (index: Int, UShort) -> Iterable\<R>)(transform): List\<R> { \
    var list = transform(index++, element)
    return List\<R>()
}

*\@SinceKotlin("1.3\")*\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\public inline fun \<R, C>
UIntArray.flatMapTo(destination: C, transform: (UInt) -> Iterable\<R>)(transform): C { \
    var index = 0
    for (element in this) {
        list = transform(index++, element)
        destination.addAll(list)
    }
    return C { \
}

*\@SinceKotlin("1.3\")*\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\public inline fun \<R, C>
UByteArray.flatMapTo(destination: C, transform: (UByte) -> Iterable\<R>)(transform): C { \
    var index = 0
    for (element in this) {
        list = transform(index++, element)
        destination.addAll(list)
    }
    return C { \
}

*\@SinceKotlin("1.3\")*\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\public inline fun \<R, C>
UShortArray.flatMapTo(destination: C, transform: (UShort) -> Iterable\<R>)(transform): C { \
    var index = 0
    for (element in this) {
        list = transform(index++, element)
        destination.addAll(list)
    }
    return C { \
}
(element in this) { val list = transform(element)
    destination.addAll(list)
} return

*An@Since Kotlin("1.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly
public inline fun <R, C : MutableCollection<in R>> ULongArray.flatMapTo(destination: C, transform: (ULong) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element)
        destination.addAll(list)
    }
    return destination
}

/**
 * Appends all elements yielded from results of [transform] function being invoked on each
 * element of original array, to the given [destination].
 */
*An@Since Kotlin("1.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly
public inline fun <R, C : MutableCollection<in R>> UByteArray.flatMapTo(destination: C, transform: (UByte) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element)
        destination.addAll(list)
    }
    return destination
}

/**
 * Appends all elements yielded from results of [transform] function being invoked on each
 * element of original array, to the given [destination].
 */
*An@Since Kotlin("1.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly
public inline fun <R, C : MutableCollection<in R>> UShortArray.flatMapTo(destination: C, transform: (UShort) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element)
        destination.addAll(list)
    }
    return destination
}

/**
 * Groups elements of the original array by the key returned by the given [keySelector]
 * function on each element and returns a map where each group key is associated with a list of
 * corresponding elements.
 */
*An@Since Kotlin("1.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly
public inline fun <K>
    UIntArray.groupBy(keySelector: (UInt) -> K): Map<K, MutableList<UInt>> {
    return groupByTo(LinkedHashMap<K, MutableList<UInt>>(), keySelector)
}

/**
 * Groups elements of the original array by the key returned by the given [keySelector]
 * function on each element and returns a map where each group key is associated with a list of
 * corresponding elements.
 */
*An@Since Kotlin("1.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly
public inline fun <K>
    ULongArray.groupBy(keySelector: (ULong) -> K): Map<K, MutableList<ULong>> {
    return groupByTo(LinkedHashMap<K, MutableList<ULong>>(), keySelector)
}

/**
 * Groups elements of the original array by the key returned by the given [keySelector]
 * function on each element and returns a map where each group key is associated with a list of
 * corresponding elements.
 */
*An@Since Kotlin("1.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly
public inline fun <K>
    UByteArray.groupBy(keySelector: (UByte) -> K): Map<K, MutableList<UByte>> {
    return groupByTo(LinkedHashMap<K, MutableList<UByte>>(), keySelector)
}

/**
 * Groups elements of the original array by the key returned by the given [keySelector]
 * function on each element and returns a map where each group key is associated with a list of
 * corresponding elements.
 */
*An@Since Kotlin("1.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly
public inline fun <K>
    UShortArray.groupBy(keySelector: (UShort) -> K): Map<K, MutableList<UShort>> {
    return groupByTo(LinkedHashMap<K, MutableList<UShort>>(), keySelector)
}
UIntArray.groupBy(keySelector: (UInt) -> K, valueTransform: (UInt) -> V): Map<K, List<V>> {
    return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)
}

/**
 * Groups values returned by the [valueTransform] function applied to each element of the original array
 * by the key returned by the given [keySelector] function applied to the element
 * and returns a map where each group key is associated with a list of corresponding values.
 * The returned map preserves the entry iteration order of the keys produced from the original array.
 */
@SinceKotlin("1.3")
public inline fun <K, V> ULongArray.groupBy(keySelector: (ULong) -> K, valueTransform: (ULong) -> V): Map<K, List<V>> {
    return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)
}

/**
 * Groups values returned by the [valueTransform] function applied to each element of the original array
 * by the key returned by the given [keySelector] function applied to the element
 * and returns a map where each group key is associated with a list of corresponding values.
 * The returned map preserves the entry iteration order of the keys produced from the original array.
 */
@SinceKotlin("1.3")
public inline fun <K, V> UByteArray.groupBy(keySelector: (UByte) -> K, valueTransform: (UByte) -> V): Map<K, List<V>> {
    return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)
}

/**
 * Groups values returned by the [valueTransform] function applied to each element of the original array
 * by the key returned by the given [keySelector] function applied to the element
 * and returns a map where each group key is associated with a list of corresponding values.
 * The returned map preserves the entry iteration order of the keys produced from the original array.
 */
@SinceKotlin("1.3")
public inline fun <K, V> UShortArray.groupBy(keySelector: (UShort) -> K, valueTransform: (UShort) -> V): Map<K, List<V>> {
    return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)
}
[destination] map each group key associated with a list of corresponding elements.

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun <K, M : MutableMap<in K, MutableList<in UShort>>> UShortArray.groupByTo(destination: M, keySelector: (UShort) -> K): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<UShort>() }
        list.add(element)
    }
    return destination
}

/**
 * Groups values returned by the [valueTransform] function applied to each element of the original array
 * by the key returned by the given [keySelector] function applied to the element
 * and puts to the [destination] map each group key associated with a list of corresponding values.
 * @return The [destination] map.
 * @sample samples.collections.Collections.Transformations.groupBy
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@kotlin.internal.InlineOnly
@public inline fun <K, V, M : MutableMap<in K, MutableList<V>>> UIntArray.groupByTo(destination: M, keySelector: (UInt) -> K, valueTransform: (UInt) -> V): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<V>() }
        list.add(valueTransform(element))
    }
    return destination
}
```

```kotlin
/**
 * Groups values returned by the [valueTransform] function applied to each element of the original array
 * by the key returned by the given [keySelector] function applied to the element
 * and puts to the [destination] map each group key associated with a list of corresponding values.
 * @return The [destination] map.
 * @sample samples.collections.Collections.Transformations.groupBy
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@kotlin.internal.InlineOnly
@public inline fun <K, V, M : MutableMap<in K, MutableList<V>>> ULongArray.groupByTo(destination: M, keySelector: (ULong) -> K, valueTransform: (ULong) -> V): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<V>() }
        list.add(valueTransform(element))
    }
    return destination
}
```

```kotlin
/**
 * Groups values returned by the [valueTransform] function applied to each element of the original array
 * by the key returned by the given [keySelector] function applied to the element
 * and puts to the [destination] map each group key associated with a list of corresponding values.
 * @return The [destination] map.
 * @sample samples.collections.Collections.Transformations.groupBy
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@kotlin.internal.InlineOnly
@public inline fun <K, V, M : MutableMap<in K, MutableList<V>>> UByteArray.groupByTo(destination: M, keySelector: (UByte) -> K, valueTransform: (UByte) -> V): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<V>() }
        list.add(valueTransform(element))
    }
    return destination
}
```

```kotlin
/**
 * Returns a list containing the results of applying the given [transform] function
 * to each element in the original array.
 * @return The [destination] map.
 * @sample samples.collections.Collections.Transformations.map
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun <R> UIntArray.map(transform: (UInt) -> R): List<R> {
    return mapTo(ArrayList<R>(size), transform)
}
```

```kotlin
/**
 * Returns a list containing the results of applying the given [transform] function
 * to each element in the original array.
 * @return The [destination] map.
 * @sample samples.collections.Collections.Transformations.map
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun <R> ULongArray.map(transform: (ULong) -> R): List<R> {
    return mapTo(ArrayList<R>(size), transform)
}
```

```kotlin
/**
 * Returns a list containing the results of applying the given [transform] function
 * to each element in the original array.
 * @return The [destination] map.
 * @sample samples.collections.Collections.Transformations.map
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun <R> UByteArray.map(transform: (UByte) -> R): List<R> {
    return mapTo(ArrayList<R>(size), transform)
}
```
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R>
UByteArray.map(transform: (UByte) -> R): List<R> {
    return mapTo(ArrayList<R>(size), transform)
}

/**
 * Returns a list containing the results of applying the given [transform] function
 * to each element in the original array.
 * @param [transform] function that takes the index of an element and the element itself
 * and returns the result of the transform applied to the element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R>
UShortArray.map(transform: (UShort) -> R): List<R> {
    return mapTo(ArrayList<R>(size), transform)
}

/**
 * Returns a list containing the results of applying the given [transform] function
 * to each element and its index in the original array.
 * @param [transform] function that takes the index of an element and the element itself
 * and returns the result of the transform applied to the element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R>
UIntArray.mapIndexed(transform: (index: Int, UInt) -> R): List<R> {
    return mapIndexedTo(ArrayList<R>(size), transform)
}

/**
 * Returns a list containing the results of applying the given [transform] function
 * to each element and its index in the original array.
 * @param [transform] function that takes the index of an element and the element itself
 * and returns the result of the transform applied to the element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R>
ULongArray.mapIndexed(transform: (index: Int, ULong) -> R): List<R> {
    return mapIndexedTo(ArrayList<R>(size), transform)
}

/**
 * Returns a list containing the results of applying the given [transform] function
 * to each element and its index in the original array.
 * @param [transform] function that takes the index of an element and the element itself
 * and returns the result of the transform applied to the element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R>
UByteArray.mapIndexed(transform: (index: Int, UByte) -> R): List<R> {
    return mapIndexedTo(ArrayList<R>(size), transform)
}

/**
 * Returns a list containing the results of applying the given [transform] function
 * to each element and its index in the original array.
 * @param [transform] function that takes the index of an element and the element itself
 * and returns the result of the transform applied to the element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R>
UShortArray.mapIndexed(transform: (index: Int, UShort) -> R): List<R> {
    return mapIndexedTo(ArrayList<R>(size), transform)
}

/**
 * Applies the given [transform] function to each element and its index in the original array
 * and appends the results to the given [destination].
 * @param [transform] function that takes the index of an element and the element itself
 * and returns the result of the transform applied to the element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R, C :
MutableCollection<in R>> UIntArray.mapIndexedTo(destination: C, transform: (index: Int, UInt) -> R): C {
    var index = 0
    for (item in this)
        destination.add(transform(index++, item))
    return destination
}

/**
 * Applies the given [transform] function to each element and its index in the original array
 * and appends the results to the given [destination].
 * @param [transform] function that takes the index of an element and the element itself
 * and returns the result of the transform applied to the element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R, C :
MutableCollection<in R>> ULongArray.mapIndexedTo(destination: C, transform: (index: Int, ULong) -> R): C {
    var index = 0
    for (item in this)
        destination.add(transform(index++, item))
    return destination
}

/**
 * Applies the given [transform] function to each element and its index in the original array
 * and appends the results to the given [destination].
 * @param [transform] function that takes the index of an element and the element itself
 * and returns the result of the transform applied to the element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R, C :
MutableCollection<in R>> UByteArray.mapIndexedTo(destination: C, transform: (index: Int, UByte) -> R): C {
    var index = 0
    for (item in this)
        destination.add(transform(index++, item))
    return destination
}

/**
 * Applies the given [transform] function to each element and its index in the original array
 * and appends the results to the given [destination].
 * @param [transform] function that takes the index of an element and the element itself
 * and returns the result of the transform applied to the element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R, C :
MutableCollection<in R>> UShortArray.mapIndexedTo(destination: C, transform: (index: Int, UShort) -> R): C {
    var index = 0
    for (item in this)
        destination.add(transform(index++, item))
    return destination
}
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R, C : MutableCollection<in R>> UShortArray.mapIndexedTo(destination: C, transform: (Int, UShort) -> R): C {
    var index = 0
    for (item in this)
        destination.add(transform(index++, item))
    return destination
}

/**
 * Applies the given [transform] function to each element of the original array
 * and appends the results to the given [destination].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R, C : MutableCollection<in R>> UIntArray.mapTo(destination: C, transform: (UInt) -> R): C {
    for (item in this)
        destination.add(transform(item))
    return destination
}

/**
 * Applies the given [transform] function to each element of the original array
 * and appends the results to the given [destination].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R, C : MutableCollection<in R>> ULongArray.mapTo(destination: C, transform: (ULong) -> R): C {
    for (item in this)
        destination.add(transform(item))
    return destination
}

/**
 * Applies the given [transform] function to each element of the original array
 * and appends the results to the given [destination].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R, C : MutableCollection<in R>> UByteArray.mapTo(destination: C, transform: (UByte) -> R): C {
    for (item in this)
        destination.add(transform(item))
    return destination
}

/**
 * Applies the given [transform] function to each element of the original array
 * and appends the results to the given [destination].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R, C : MutableCollection<in R>> UShortArray.mapTo(destination: C, transform: (UShort) -> R): C {
    for (item in this)
        destination.add(transform(item))
    return destination
}

/**
 * Returns a lazy [Iterable] that wraps each element of the original array
 * into an [IndexedValue] containing the index of that element and the element
 * itself.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray.withIndex(): Iterable<IndexedValue<UInt>> {
    return IndexingIterable { iterator() }
}

/**
 * Returns a lazy [Iterable] that wraps each element of the original array
 * into an [IndexedValue] containing the index of that element and the element
 * itself.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongArray.withIndex(): Iterable<IndexedValue<ULong>> {
    return IndexingIterable { iterator() }
}

/**
 * Returns a lazy [Iterable] that wraps each element of the original array
 * into an [IndexedValue] containing the index of that element and the element
 * itself.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.withIndex(): Iterable<IndexedValue<UByte>> {
    return IndexingIterable { iterator() }
}

/**
 * Returns a lazy [Iterable] that wraps each element of the original array
 * into an [IndexedValue] containing the index of that element and the element
 * itself.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.withIndex(): Iterable<IndexedValue<UShort>> {
    return IndexingIterable { iterator() }
}

/**
 * Returns `true` if all elements match the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.all
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.all(predicate: (UInt) -> Boolean): Boolean {
    for (element in this) if (!predicate(element)) return false
    return true
}

/**
 * Returns `true` if all elements match the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.all
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.all(predicate: (ULong) -> Boolean): Boolean {
    for (element in this) if (!predicate(element)) return false
    return true
}

/**
 * Returns `true` if all elements match the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.all
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.all(predicate: (UByte) -> Boolean): Boolean {
    for (element in this) if (!predicate(element)) return false
    return true
}

/**
 * Returns `true` if all elements match the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.all
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.all(predicate: (UShort) -> Boolean): Boolean {
    for (element in this) if (!predicate(element)) return false
    return true
}
samples.collections.Collections.Aggregates.any
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UIntArray.any(): Boolean {
\n    return storage.any()\n\n} *\n
Any other

samples.collections.Collections.Aggregates.any
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
ULongArray.any(): Boolean {
\n    return storage.any()\n\n} *\n
Any other

samples.collections.Collections.Aggregates.any
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UByteArray.any(): Boolean {
\n    return storage.any()\n\n} *\n
Any other

samples.collections.Collections.Aggregates.any
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UShortArray.any(): Boolean {
\n    return storage.any()\n\n} *\n
Any other

samples.collections.Collections.Aggregates.any
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UIntArray.any(predicate: (UInt) -> Boolean): Boolean {
\n    for (element in this) if (predicate(element)) return true
\n    return false
\n} *\n
Any other

samples.collections.Collections.Aggregates.any
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
ULongArray.any(predicate: (ULong) -> Boolean): Boolean {
\n    for (element in this) if (predicate(element)) return true
\n    return false
\n} *\n
Any other

samples.collections.Collections.Aggregates.any
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UByteArray.any(predicate: (UByte) -> Boolean): Boolean {
\n    for (element in this) if (predicate(element)) return true
\n    return false
\n} *\n
Any other

samples.collections.Collections.Aggregates.any
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UShortArray.any(predicate: (UShort) -> Boolean): Boolean {
\n    for (element in this) if (predicate(element)) return true
\n    return false
\n} *\n
Any other

samples.collections.Collections.Aggregates.any
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UIntArray.count(predicate: (UInt) -> Boolean): Int {
\n    var count = 0\n    for (element in this) if (predicate(element)) ++count
\n    return count
\n} *\n
Any other

samples.collections.Collections.Aggregates.any
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
ULongArray.count(predicate: (ULong) -> Boolean): Int {
\n    var count = 0\n    for (element in this) if (predicate(element)) ++count
\n    return count
\n} *\n
Any other

samples.collections.Collections.Aggregates.any
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UByteArray.count(predicate: (UByte) -> Boolean): Int {
\n    var count = 0\n    for (element in this) if (predicate(element)) ++count
\n    return count
\n} *\n
Any other

samples.collections.Collections.Aggregates.any
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UShortArray.count(predicate: (UShort) -> Boolean): Int {
\n    var count = 0\n    for (element in this) if (predicate(element)) ++count
\n    return count
\n} *\n
Any other

samples.collections.Collections.Aggregates.any
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UIntArray.fold(initial: R, operation: (acc: R, UInt) -> R): R {
\n    var accumulator = initial
    for (element in this) accumulator = operation(accumulator, element)
\n    return accumulator
\n} *\n
Any other
current accumulator value and an element, and calculates the next accumulator value.

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline fun <R>
ULongArray.fold(initial: R, operation: (acc: R, ULong) -> R): R {
    var accumulator = initial
    for (element in this) accumulator = operation(accumulator, element)
    return accumulator
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline fun <R>
UByteArray.fold(initial: R, operation: (acc: R, UByte) -> R): R {
    var accumulator = initial
    for (element in this) accumulator = operation(accumulator, element)
    return accumulator
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline fun <R>
    var accumulator = initial
    for (element in this) accumulator = operation(accumulator, element)
    return accumulator
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline fun <R>
UIntArray.foldIndexed(initial: R, operation: (index: Int, acc: R, UInt) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline fun <R>
ULongArray.foldIndexed(initial: R, operation: (index: Int, acc: R, ULong) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline fun <R>
UByteArray.foldIndexed(initial: R, operation: (index: Int, acc: R, UByte) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline fun <R>
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}
```

* @param [operation] function that takes an index of an element, current accumulator value and an element, and calculates the next accumulator value.

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline fun <R>\nUByteArray.foldIndexed(initial: R, operation: (index: Int, acc: R, UByte) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}
```
```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R>
UIntArray.foldRight(initial: R, operation: (UInt, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from right to left
 * to each element and current accumulator value.
 * Returns the specified [initial] value if the array is empty.
 * @param [operation] function that takes an element and current accumulator value, and calculates the next accumulator value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R>
ULongArray.foldRight(initial: R, operation: (ULong, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from right to left
 * to each element and current accumulator value.
 * Returns the specified [initial] value if the array is empty.
 * @param [operation] function that takes an element and current accumulator value, and calculates the next accumulator value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R>
UByteArray.foldRight(initial: R, operation: (UByte, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from right to left
 * to each element and current accumulator value.
 * Returns the specified [initial] value if the array is empty.
 * @param [operation] function that takes an element and current accumulator value, and calculates the next accumulator value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R>
UShortArray.foldRight(initial: R, operation: (UShort, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from right to left
 * to each element with its index in the original array and current accumulator value.
 * Returns the specified [initial] value if the array is empty.
 * @param [operation] function that takes an element, the index of an element, and current accumulator value, and calculates the next accumulator value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R>
UIntArray.foldRightIndexed(initial: R, operation: (index: Int, UInt, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from right to left
 * to each element with its index in the original array and current accumulator value.
 * Returns the specified [initial] value if the array is empty.
 * @param [operation] function that takes an element, the index of an element, and current accumulator value, and calculates the next accumulator value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R>
ULongArray.foldRightIndexed(initial: R, operation: (index: Int, ULong, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from right to left
 * to each element with its index in the original array and current accumulator value.
 * Returns the specified [initial] value if the array is empty.
 * @param [operation] function that takes an element, the index of an element, and current accumulator value, and calculates the next accumulator value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R>
UByteArray.foldRightIndexed(initial: R, operation: (index: Int, UByte, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from right to left
 * to each element with its index in the original array and current accumulator value.
 * Returns the specified [initial] value if the array is empty.
 * @param [operation] function that takes an element, the index of an element, and current accumulator value, and calculates the next accumulator value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R>
UShortArray.foldRightIndexed(initial: R, operation: (index: Int, UShort, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

```
accumulator value.\n* \n* Returns the specified [initial] value if the array is empty.\n* \n@param [operation] function that takes the index of an element, the element itself\n* and current accumulator value, and calculates the next accumulator value.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R>\nUSHortArray.foldRightIndexed(initial: R, operation: (Int, UShort, acc: R) -> R): R { \n    var index = \n    var accumulator = initial while (index >= 0) { \n        accumulator = operation(index, get(index), accumulator) \n        --index \n    } \n    return accumulator
}

*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.forEach(action: (UInt) -> Unit): Unit { \n    for (element in this) action(element)
}

*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.forEachIndexed(action: (Int, ULong) -> Unit): Unit { \n    var index = 0 \n    for (item in this) action(index++, item)
}

*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.forEachIndexed(action: (Int, UByte) -> Unit): Unit { \n    var index = 0 \n    for (item in this) action(index++, item)
}

*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.forEachIndexed(action: (Int, UShort) -> Unit): Unit { \n    var index = 0 \n    for (item in this) action(index++, item)
}

@Deprecated("Use maxOrNull instead.",\nReplaceWith("this.maxOrNull()"))\n@DeprecatedSinceKotlin(warningSince = "1.4")\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.max(): UInt? { \n    return maxOrNull()
}

@Deprecated("Use maxOrNull instead.",\nReplaceWith("this.maxOrNull()"))\n@DeprecatedSinceKotlin(warningSince = "1.4")\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.max(): ULong? { \n    return maxOrNull()
}

@Deprecated("Use maxOrNull instead.",\nReplaceWith("this.maxOrNull()"))\n@DeprecatedSinceKotlin(warningSince = "1.4")\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.max(): UByte? { \n    return maxOrNull()
}

@Deprecated("Use maxOrNull instead.",\nReplaceWith("this.maxOrNull()"))\n@DeprecatedSinceKotlin(warningSince = "1.4")\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.max(): UShort? { \n    return maxOrNull()
}
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@Suppress("UnstableApi")
public fun UShortArray.max(): UShort? {
    return maxOrNull()
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> UIntArray.maxBy(selector: (UInt) -> R): UInt? {
    return maxByOrNull(selector)
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> ULongArray.maxBy(selector: (ULong) -> R): ULong? {
    return maxByOrNull(selector)
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> UByteArray.maxBy(selector: (UByte) -> R): UByte? {
    return maxByOrNull(selector)
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> UShortArray.maxBy(selector: (UShort) -> R): UShort? {
    return maxByOrNull(selector)
}

/**
 * Returns the first element yielding the largest value of the given function or `null` if there are no elements.
 * 
 * @sample samples.collections.Collections.Aggregates.max
 */
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> UIntArray.maxByOrNull(selector: (UInt) -> R): UInt? {
    if (isEmpty()) return null
    var maxElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxValue = selector(maxElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    }
    return maxElem
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array.
 * 
 * @throws NoSuchElementException if the array is empty.
 */
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> ULongArray.maxByOrNull(selector: (ULong) -> R): ULong? {
    if (isEmpty()) return null
    var maxElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxValue = selector(maxElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    }
    return maxElem
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array.
 * 
 * @throws NoSuchElementException if the array is empty.
 */
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> UByteArray.maxByOrNull(selector: (UByte) -> R): UByte? {
    if (isEmpty()) return null
    var maxElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxValue = selector(maxElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    }
    return maxElem
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array.
 * 
 * @throws NoSuchElementException if the array is empty.
 */
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> UShortArray.maxByOrNull(selector: (UShort) -> R): UShort? {
    if (isEmpty()) return null
    var maxElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxValue = selector(maxElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    }
    return maxElem
}

/**
 * Returns the first element yielding the largest value of the given function or `null` if there are no elements.
 * 
 * @sample samples.collections.Collections.Aggregates.max
 */
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> UByteArray.maxByOrNull(selector: (UByte) -> R): UByte? {
    if (isEmpty()) return null
    var maxElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxValue = selector(maxElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    }
    return maxElem
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array.
 * 
 * @throws NoSuchElementException if the array is empty.
 */
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> UCharArray.maxByOrNull(selector: (UChar) -> R): UChar? {
    if (isEmpty()) return null
    var maxElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxValue = selector(maxElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    }
    return maxElem
}
/**
 * Returns the largest value among all values produced
 * by [selector] function\n * applied to each element in the array.\n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * @throws NoSuchElementException if the array is empty.\n */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
UIntArray.maxOf(selector: (UInt) -> Double): Double {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced
 * by [selector] function\n * applied to each element in the array.\n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * @throws NoSuchElementException if the array is empty.\n */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
ULongArray.maxOf(selector: (ULong) -> Double): Double {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced
 * by [selector] function\n * applied to each element in the array.\n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * @throws NoSuchElementException if the array is empty.\n */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
UByteArray.maxOf(selector: (UByte) -> Double): Double {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced
 * by [selector] function\n * applied to each element in the array.\n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * @throws NoSuchElementException if the array is empty.\n */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
UShortArray.maxOf(selector: (UShort) -> Double): Double {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced
 * by [selector] function\n * applied to each element in the array.\n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * @throws NoSuchElementException if the array is empty.\n */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
UIntArray.maxOf(selector: (UInt) -> Float): Float {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced
 * by [selector] function\n * applied to each element in the array.\n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * @throws NoSuchElementException if the array is empty.\n */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
ULongArray.maxOf(selector: (ULong) -> Float): Float {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced
 * by [selector] function\n * applied to each element in the array.\n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * @throws NoSuchElementException if the array is empty.\n */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
UByteArray.maxOf(selector: (UByte) -> Float): Float {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced
 * by [selector] function\n * applied to each element in the array.\n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * @throws NoSuchElementException if the array is empty.\n */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
UShortArray.maxOf(selector: (UShort) -> Float): Float {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}
by [selector] function\n* applied to each element in the array.\n* \n* \n* If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n* \n* \n* @throws NoSuchElementException if the array is empty.\n* \n* \n* @SinceKotlin("1.4")\n* @OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n* @OverloadResolution\n* @ResponseBody\n* @ExperimentalUnsignedTypes\n* @kotlin.internal.InlineOnly\n* public inline fun UShortArray.maxOf(selector: (UShort) -> Float): Float {\n    \n    \n    \n    \n} \n\n* \n* \n* @SinceKotlin("1.4")\n* @OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n* @OverloadResolution\n* ByLambdaReturnType\n* @ExperimentalUnsignedTypes\n* @kotlin.internal.InlineOnly\n* public inline fun UShortArray.maxOfOrNull(selector: (UShort) -> Double): Double? {\n    \n    \n    \n    \n} \n\n* \n* \n* @SinceKotlin("1.4")\n* @OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n* @OverloadResolution\n* ByLambdaReturnType\n* @ExperimentalUnsignedTypes\n* @kotlin.internal.InlineOnly\n* public inline fun UIntArray.maxOfOrNull(selector: (UInt) -> Double): Double? {\n    \n    \n    \n    \n} \n\n* \n* \n* @SinceKotlin("1.4")\n* @OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n* @OverloadResolution\n* ByLambdaReturnType\n* @ExperimentalUnsignedTypes\n* @kotlin.internal.InlineOnly\n* public inline fun ULongArray.maxOfOrNull(selector: (ULong) -> Double): Double? {\n    \n    \n    \n    \n} \n\n* \n* \n* @SinceKotlin("1.4")\n* @OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n* @OverloadResolution\n* ByByteArray.maxOfOrNull(selector: (UByte) -> Double): Double? {\n    \n    \n    \n    \n} \n\n* \n* \n* @SinceKotlin("1.4")\n* @OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n* @OverloadResolution\n* ByByteArray.maxOfOrNull(selector: (UByte) -> Double): Double? {\n    \n    \n    \n    \n}
maxValue = selector(this[0])
for (i in 1..lastIndex) {
    val v = selector(this[i])
    maxValue = maxOf(maxValue, v)
}
return maxValue

@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.maxOfOrNull(selector: (UByte) -> Double): Double? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.maxOfOrNull(selector: (UShort) -> Double): Double? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.maxOfOrNull(selector: (UInt) -> Float): Float? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.maxOfOrNull(selector: (ULong) -> Float): Float? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> UIntArray.maxOfOrNull(selector: (UInt) -> R): R? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.maxOfOrNull(selector: (UByte) -> Float): Float? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.maxOfOrNull(selector: (UShort) -> Float): Float? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> UByteArray.maxOfOrNull(selector: (UByte) -> R): R? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> UShortArray.maxOfOrNull(selector: (UShort) -> R): R? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}
maxValue = selector(this[0])
for (i in 1..lastIndex) {
    val v = selector(this[i])
    if (maxValue < v) {
        maxValue = v
    }
}
return maxValue

/**
 * Returns the largest value among all values
produced by [selector] function* applied to each element in the array or 'null' if there are no elements.
*/
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> ULongArray.maxOfOrNull(selector: (ULong) -> R): R? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value among all values
produced by [selector] function* applied to each element in the array or 'null' if there are no elements.
*/
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> UByteArray.maxOfOrNull(selector: (UByte) -> R): R? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value among all values
produced by [selector] function* applied to each element in the array or 'null' if there are no elements.
*/
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> UShortArray.maxOfOrNull(selector: (UShort) -> R): R? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value according to the provided [comparator]
* among all values produced by [selector] function applied to each element in the array.
*/
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : Comparator<in R>> UIntArray.maxOfWith(comparator: Comparator<in R>, selector: (UInt) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(maxValue, v) < 0) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value according to the provided [comparator]
* among all values produced by [selector] function applied to each element in the array.
*/
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : Comparator<in R>> ULongArray.maxOfWith(comparator: Comparator<in R>, selector: (ULong) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(maxValue, v) < 0) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value according to the provided [comparator]
* among all values produced by [selector] function applied to each element in the array.
*/
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : Comparator<in R>> UByteArray.maxOfWith(comparator: Comparator<in R>, selector: (UByte) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(maxValue, v) < 0) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value according to the provided [comparator]
* among all values produced by [selector] function applied to each element in the array.
*/
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : Comparator<in R>> UShortArray.maxOfWith(comparator: Comparator<in R>, selector: (UShort) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(maxValue, v) < 0) {
            maxValue = v
        }
    }
    return maxValue
}
UShortArray.maxOfWith(comparator: Comparator<in R>, selector: (UShort) -> R): R
    \{ n \ if (isEmpty()) throw
    NoSuchElementException() \n    var maxValue = selector(this[0]) \n    for (i in 1..lastIndex) \n    \{ n val v = selector(this[i]) \n    if (comparator.compare(maxValue, v) < 0) \n    \{ n maxValue = v \n    \} n return maxValue
    \} n/**n * Returns the largest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.

*\An@SinceKotlin("1.4")\u@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\u@OverloadResolution
ByLambdaReturnType\u@ExperimentalUnsignedTypes\u@kotlin.internal.InlineOnly\u@public inline fun <R>
UShortArray.maxOfOrNull(comparator: Comparator<in R>, selector: (UShort) -> R): R?
    \{ n \ if (isEmpty()) return null
    var maxValue = selector(this[0]) \n    for (i in 1..lastIndex) \n    \{ n val v = selector(this[i]) \n    if (comparator.compare(maxValue, v) < 0) \n    \{ n maxValue = v \n    \} n return maxValue
    \} n/**n * Returns the largest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.

*\An@SinceKotlin("1.4")\u@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\u@OverloadResolution
ByLambdaReturnType\u@ExperimentalUnsignedTypes\u@kotlin.internal.InlineOnly\u@public inline fun <R>
ULongArray.maxOfOrNull(comparator: Comparator<in R>, selector: (ULong) -> R): ULong?
    \{ n \ if (isEmpty()) return null
    var maxValue = selector(this[0]) \n    for (i in 1..lastIndex) \n    \{ n val v = selector(this[i]) \n    if (comparator.compare(maxValue, v) < 0) \n    \{ n maxValue = v \n    \} n return maxValue
    \} n/**n * Returns the largest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.

*\An@SinceKotlin("1.4")\u@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\u@OverloadResolution
ByLambdaReturnType\u@ExperimentalUnsignedTypes\u@kotlin.internal.InlineOnly\u@public inline fun <R>
UByteArray.maxOfOrNull(comparator: Comparator<in R>, selector: (UByte) -> R): UByte?
    \{ n \ if (isEmpty()) return null
    var maxValue = selector(this[0]) \n    for (i in 1..lastIndex) \n    \{ n val v = selector(this[i]) \n    if (comparator.compare(maxValue, v) < 0) \n    \{ n maxValue = v \n    \} n return maxValue
    \} n/**n * Returns the largest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.

*\An@SinceKotlin("1.4")\u@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\u@OverloadResolution
ByLambdaReturnType\u@ExperimentalUnsignedTypes\u@kotlin.internal.InlineOnly\u@public inline fun <R>
UShortArray.maxOfOrNull(comparator: Comparator<in R>, selector: (UShort) -> R): UShort?
    \{ n \ if (isEmpty()) return null
    var maxValue = selector(this[0]) \n    for (i in 1..lastIndex) \n    \{ n val v = selector(this[i]) \n    if (comparator.compare(maxValue, v) < 0) \n    \{ n maxValue = v \n    \} n return maxValue
    \} n/**n * Returns the largest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.

*\An@SinceKotlin("1.4")\u@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\u@OverloadResolution
ByLambdaReturnType\u@ExperimentalUnsignedTypes\u@kotlin.internal.InlineOnly\u@public inline fun UIntArray.maxOrNull(): UInt?
    \{ n \ if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) \n    \{ n e = this[i]
    if (max < e) max = e
    \} n return max
    \} n/**n * Returns the largest element or `null` if there are no elements.

*\An@SinceKotlin("1.4")\u@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\u@OverloadResolution
ByLambdaReturnType\u@ExperimentalUnsignedTypes\u@kotlin.internal.InlineOnly\u@public inline fun ULongArray.maxOrNull(): ULong?
    \{ n \ if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) \n    \{ n e = this[i]
    if (max < e) max = e
    \} n return max
    \} n/**n * Returns the largest element or `null` if there are no elements.

*\An@SinceKotlin("1.4")\u@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\u@OverloadResolution
ByLambdaReturnType\u@ExperimentalUnsignedTypes\u@kotlin.internal.InlineOnly\u@public inline fun UByteArray.maxOrNull(): UByte?
    \{ n \ if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) \n    \{ n e = this[i]
    if (max < e) max = e
    \} n return max
    \} n/**n * Returns the largest element or `null` if there are no elements.

*\An@SinceKotlin("1.4")\u@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\u@OverloadResolution
ByLambdaReturnType\u@ExperimentalUnsignedTypes\u@kotlin.internal.InlineOnly\u@public inline fun UShortArray.maxOrNull(): UShort?
    \{ n \ if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) \n    \{ n e = this[i]
    if (max < e) max = e
    \} n return max
    \} n/**n * Returns the largest element or `null` if there are no elements.

@Deprecated("Use maxWithOrNull instead.",
ReplaceWith("this.maxWithOrNull(comparator)"))\u@DeprecatedSinceKotlin(warningSince =
"1.4")\u@SinceKotlin("1.3")\u@ExperimentalUnsignedTypes\u@public inline fun UIntArray.maxWith(comparator:
Comparator<in UInt>): UInt?
    \{ n return maxWithOrNull(comparator)
    \} n/**n * Returns the largest element or `null` if there are no elements.

@Deprecated("Use maxWithOrNull instead.",
ReplaceWith("this.maxWithOrNull(comparator)"))\u@DeprecatedSinceKotlin(warningSince =
/**
 * Returns the first element having the largest value according to the provided 
 * comparator or `null` if there are no elements.
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
public fun ULongArray.maxWithOrNull(comparator:
    Comparator<in ULong>): ULong? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(max, e) < 0) max = e
    }
    return max
}

/**
 * Returns the first element having the largest value according to the provided 
 * comparator or `null` if there are no elements.
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
public fun UByteArray.maxWithOrNull(comparator:
    Comparator<in UByte>): UByte? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(max, e) < 0) max = e
    }
    return max
}

/**
 * Returns the first element having the largest value according to the provided 
 * comparator or `null` if there are no elements.
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
public fun UShortArray.maxWithOrNull(comparator:
    Comparator<in UShort>): UShort? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(max, e) < 0) max = e
    }
    return max
}

@Deprecated("Use minOrNull instead.
    ReplaceWith("this.minOrNull()")")
@DeprecatedSinceKotlin(warningSince =
    \"1.4\")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.min(): UByte? {
    return minOrNull()
}

@Deprecated("Use minOrNull instead.
    ReplaceWith("this.minOrNull()")")
@DeprecatedSinceKotlin(warningSince =
    \"1.4\")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.min(): UShort? {
    return minOrNull()
}

@Deprecated("Use minByOrNull instead.
    ReplaceWith("this.minByOrNull(selector)")")
@DeprecatedSinceKotlin(warningSince =
    \"1.4\")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray.minBy(selector: (UInt) -> R): UInt? {
    return minByOrNull(selector)
}

@Deprecated("Use minByOrNull instead.
    ReplaceWith("this.minByOrNull(selector)")")
@DeprecatedSinceKotlin(warningSince =
    \"1.4\")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongArray.minBy(selector: (ULong) -> R): ULong? {
    return minByOrNull(selector)
}

@Deprecated("Use minByOrNull instead.
    ReplaceWith("this.minByOrNull(selector)")")
@DeprecatedSinceKotlin(warningSince =
    \"1.4\")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.minBy(selector: (UByte) -> R): UByte? {
    return minByOrNull(selector)
}

@Deprecated("Use minByOrNull instead.
    ReplaceWith("this.minByOrNull(selector)")")
@DeprecatedSinceKotlin(warningSince =
    \"1.4\")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.minBy(selector: (UShort) -> R): UShort? {
    return minByOrNull(selector)
}

@Deprecated("Use minByOrNull instead.
    ReplaceWith("this.minByOrNull(selector)")")
@DeprecatedSinceKotlin(warningSince =
    \"1.4\")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray.minByOrNull(selector: (UInt) -> R): UInt? {
    return minByOrNull(selector)
}

@Deprecated("Use minByOrNull instead.
    ReplaceWith("this.minByOrNull(selector)")")
@DeprecatedSinceKotlin(warningSince =
    \"1.4\")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.minByOrNull(selector: (UByte) -> R): UByte? {
    return minByOrNull(selector)
}

@Deprecated("Use minByOrNull instead.
    ReplaceWith("this.minByOrNull(selector)")")
@DeprecatedSinceKotlin(warningSince =
    \"1.4\")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.minByOrNull(selector: (UShort) -> R): UShort? {
    return minByOrNull(selector)
}

@Deprecated("Use minByOrNull instead.
    ReplaceWith("this.minByOrNull(selector)")")
@DeprecatedSinceKotlin(warningSince =
    \"1.4\")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongArray.minByOrNull(selector: (ULong) -> R): ULong? {
    return minByOrNull(selector)
}

@Deprecated("Use minByOrNull instead.
    ReplaceWith("this.minByOrNull(selector)")")
@DeprecatedSinceKotlin(warningSince =
    \"1.4\")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray.minByOrNull(selector: (UInt) -> R): UInt? {
    return minByOrNull(selector)
}

@Deprecated("Use minByOrNull instead.
    ReplaceWith("this.minByOrNull(selector)")")
@DeprecatedSinceKotlin(warningSince =
    \"1.4\")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.minByOrNull(selector: (UByte) -> R): UByte? {
    return minByOrNull(selector)
}

@Deprecated("Use minByOrNull instead.
    ReplaceWith("this.minByOrNull(selector)")")
@DeprecatedSinceKotlin(warningSince =
    \"1.4\")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.minByOrNull(selector: (UShort) -> R): UShort? {
    return minByOrNull(selector)
}

@Deprecated("Use minByOrNull instead.
    ReplaceWith("this.minByOrNull(selector)")")
@DeprecatedSinceKotlin(warningSince =
    \"1.4\")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongArray.minByOrNull(selector: (ULong) -> R): ULong? {
    return minByOrNull(selector)
}

@Deprecated("Use minByOrNull instead.
    ReplaceWith("this.minByOrNull(selector)")")
@DeprecatedSinceKotlin(warningSince =
    \"1.4\")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray.minByOrNull(selector: (UInt) -> R): UInt? {
    return minByOrNull(selector)
}

@Deprecated("Use minByOrNull instead.
    ReplaceWith("this.minByOrNull(selector)")")
@DeprecatedSinceKotlin(warningSince =
    \"1.4\")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.minByOrNull(selector: (UByte) -> R): UByte? {
    return minByOrNull(selector)
}

@Deprecated("Use minByOrNull instead.
    ReplaceWith("this.minByOrNull(selector)")")
@DeprecatedSinceKotlin(warningSince =
    \"1.4\")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.minByOrNull(selector: (UShort) -> R): UShort? {
    return minByOrNull(selector)
)

@Deprecated("Use minByOrNull instead.
    ReplaceWith("this.minByOrNull(selector)")")
@DeprecatedSinceKotlin(warningSince =
    \"1.4\")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongArray.minByOrNull(selector: (ULong) -> R): ULong? {
    return minByOrNull(selector)
)

@Deprecated("Use minByOrNull instead.
    ReplaceWith("this.minByOrNull(selector)")")
@DeprecatedSinceKotlin(warningSince =
    \"1.4\")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray.minByOrNull(selector: (UInt) -> R): UInt? {
    return minByOrNull(selector)
)

@Deprecated("Use minByOrNull instead.
    ReplaceWith("this.minByOrNull(selector)")")
@DeprecatedSinceKotlin(warningSince =
    \"1.4\")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.minByOrNull(selector: (UByte) -> R): UByte? {
    return minByOrNull(selector)
)

@Deprecated("Use minByOrNull instead.
    ReplaceWith("this.minByOrNull(selector)")")
@DeprecatedSinceKotlin(warningSince =
    \"1.4\")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.minByOrNull(selector: (UShort) -> R): UShort? {
    return minByOrNull(selector)
)
"1.4")\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> UByteArray.minBy(selector: (UByte) -> R): UByte? {
    return minByOrNull(selector)\n}\n@Deprecated("Use minByOrNull instead.", ReplaceWith("this.minByOrNull(selector())"))\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> UShortArray.minBy(selector: (UShort) -> R): UShort? {
    return minByOrNull(selector)\n}\n@SinceKotlin("1.4")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> ULongArray.minBy(selector: (ULong) -> R): ULong? {
    return minByOrNull(selector)\n}\n@SinceKotlin("1.4")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> UIntArray.minByOrNull(selector: (UInt) -> R): UInt? {
    if (isEmpty()) return null\n    var minElem = this[0]\n    val lastIndex = this.lastIndex\n    if (lastIndex == 0) return minElem\n    var minValue = selector(minElem) for (i in 1..lastIndex) {\n        val e = this[i]\n        val v = selector(e) if (minValue > v) {
            minElem = e\n            minValue = v
        }
    }\n    return minElem
}

/**
 * Returns the first element yielding the smallest value of the given function or `null` if there are no elements.
 * 
 * @sample samples.collections.Collections.Aggregates.minByOrNull
*/
@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.minOf(selector: (UInt) -> Double): Double {
    if (isEmpty()) throw NoSuchElementException()\n    var minValue = selector(this[0])\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        val v = selector(e) if (minValue > v) {
            minValue = v
        }
    }\n    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array.\n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * @throws NoSuchElementException if the array is empty.\n*/
```kotlin
minOf(minValue, v) \n |n return minValue\n}n\n/**\n * Returns the smallest value among all values produced
 * by [selector] function\n \n * Any values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * @throws NoSuchElementException if the array is empty.\n *@\n */\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun\nUByteArray.minOf(selector: (UByte) -> Double): Double \n|n if (isEmpty()) throw NoSuchElementException()\nvar minValue = selector(this[0])\nfor (i in 1...lastIndex) \n|n val v = selector(this[i])\n\nminValue = minOf(minValue, v) \n\n|n return minValue\n}n\n/**\n * Returns the smallest value among all values produced
 * by [selector] function\n \n * Any values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * @throws NoSuchElementException if the array is empty.\n *@\n */\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun\nUShortArray.minOf(selector: (UShort) -> Double): Double \n|n if (isEmpty()) throw NoSuchElementException()\nvar minValue = selector(this[0])\nfor (i in 1...lastIndex) \n|n val v = selector(this[i])\n\nminValue = minOf(minValue, v) \n\n|n return minValue\n}n\n/**\n * Returns the smallest value among all values produced
 * by [selector] function\n \n * Any values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * @throws NoSuchElementException if the array is empty.\n *@\n */\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun\nUIntArray.minOf(selector: (UInt) -> Float): Float \n|n if (isEmpty()) throw NoSuchElementException()\nvar minValue = selector(this[0])\nfor (i in 1...lastIndex) \n|n val v = selector(this[i])\n\nminValue = minOf(minValue, v) \n\n|n return minValue\n}n\n/**\n * Returns the smallest value among all values produced
 * by [selector] function\n \n * Any values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * @throws NoSuchElementException if the array is empty.\n *@\n */\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun\nULongArray.minOf(selector: (ULong) -> Float): Float \n|n if (isEmpty()) throw NoSuchElementException()\nvar minValue = selector(this[0])\nfor (i in 1...lastIndex) \n|n val v = selector(this[i])\n\nminValue = minOf(minValue, v) \n\n|n return minValue\n}n\n/**\n * Returns the smallest value among all values produced
 * by [selector] function\n \n * Any values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * @throws NoSuchElementException if the array is empty.\n *@\n */\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun\nUByteArray.minOf(selector: (UByte) -> Float): Float \n|n if (isEmpty()) throw NoSuchElementException()\nvar minValue = selector(this[0])\nfor (i in 1...lastIndex) \n|n val v = selector(this[i])\n\nminValue = minOf(minValue, v) \n\n|n return minValue\n}n\n/**\n * Returns the smallest value among all values produced
 * by [selector] function\n \n * Any values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * @throws NoSuchElementException if the array is empty.\n *@\n */\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun\nUShortArray.minOf(selector: (UShort) -> Float): Float \n|n if (isEmpty()) throw NoSuchElementException()\nvar minValue = selector(this[0])\nfor (i in 1...lastIndex) \n|n val v = selector(this[i])\n\nminValue = minOf(minValue, v) \n\n|n return minValue\n}n\n/**\n * Returns the smallest value among all values produced
 * by [selector] function\n \n * Any values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * @throws NoSuchElementException if the array is empty.\n *@\n */\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun \n```
NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the array.
 * @throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> ULongArray.minOf(selector: (ULong) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the array.
 * @throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : UByte> UByteArray.minOf(selector: (UByte) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the array.
 * @throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : UShort> UShortArray.minOf(selector: (UShort) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 * If any of values produced by [selector] function is `NaN`, the returned
 * result is `NaN`.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.minOfOrNull(selector: (UInt) -> Double): Double? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 * If any of values produced by [selector] function is `NaN`, the returned
 * result is `NaN`.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.minOfOrNull(selector: (ULong) -> Double): Double? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 * If any of values produced by [selector] function is `NaN`, the returned
 * result is `NaN`.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.minOfOrNull(selector: (UByte) -> Double): Double? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}
ByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UShortArray.minOfOrNull(selector: (UShort) -> Double): Double? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
} // Returns the smallest value among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.

*\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UIntArray.minOfOrNull(selector: (UInt) -> Float): Float? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
} // Returns the smallest value among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.

*\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
ULongArray.minOfOrNull(selector: (ULong) -> Float): Float? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
} // Returns the smallest value among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.

*\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UByteArray.minOfOrNull(selector: (UByte) -> Float): Float? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
} // Returns the smallest value among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.

*\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
<R : Comparable<R>> UIntArray.minOfOrNull(selector: (UInt) -> R): R? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
} // Returns the smallest value among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.

*\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
<R : Comparable<R>> ULongArray.minOfOrNull(selector: (ULong) -> R): R? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
} // Returns the smallest value among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.

*\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
<R : Comparable<R>> UByteArray.minOfOrNull(selector: (UByte) -> R): R? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
} // Returns the smallest value among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.
minValue = selector(this[0])
for (i in 1..lastIndex) {
    val v = selector(this[i])
    if (minValue > v) {
        minValue = v
    }
}

/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> UShortArray.minOfOrNull(selector: (UShort) -> R): R? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> UShortArray.minOfWith(comparator: Comparator<R>, selector: (UShort) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(minValue, v) > 0) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> UShortArray.minOfWithOrNull(comparator: Comparator<R>, selector: (UShort) -> R): R? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(minValue, v) > 0) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> UShortArray.minOfWith(comparator: Comparator<R>, selector: (UShort) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(minValue, v) > 0) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> UShortArray.minOfWithOrNull(comparator: Comparator<R>, selector: (UShort) -> R): R? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(minValue, v) > 0) {
            minValue = v
        }
    }
    return minValue
}
/*\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R>\nULongArray.minOfWithOrNull(comparator: Comparator<in R>, selector: (ULong) -> R): R? {
    if (isEmpty()) return null\n    var minValue = selector(this[0])\n    for (i in 1..lastIndex) {
        val v = selector(this[i])\n        if (comparator.compare(minValue, v) > 0) {
            minValue = v
        }
    }
    return minValue
}*

/**
* Returns the smallest value according to the provided [comparator]*\n* among all values produced by [selector] function applied to each element in the array or ‘null’ if there are no elements.*\n*/
@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R>\nUByteArray.minOfWithOrNull(comparator: Comparator<in R>, selector: (UByte) -> R): R? {
    if (isEmpty()) return null\n    var minValue = selector(this[0])\n    for (i in 1..lastIndex) {
        val v = selector(this[i])\n        if (comparator.compare(minValue, v) > 0) {
            minValue = v
        }
    }
    return minValue
}*

/**
* Returns the smallest value according to the provided [comparator]*\n* among all values produced by [selector] function applied to each element in the array or ‘null’ if there are no elements.*\n*/
@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R>\nUShortArray.minOfWithOrNull(comparator: Comparator<in R>, selector: (UShort) -> R): R? {
    if (isEmpty()) return null\n    var minValue = selector(this[0])\n    for (i in 1..lastIndex) {
        val v = selector(this[i])\n        if (comparator.compare(minValue, v) > 0) {
            minValue = v
        }
    }
    return minValue
}*

/**
* Returns the first element\n* having the smallest value according to the provided [comparator] or ‘null’ if there are no elements.*\n*/
@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic fun UIntArray.minOrNull(): UInt? {
    if (isEmpty()) return null\n    var min = this[0]\n    for (i in 1..lastIndex) {
        val e = this[i]\n        if (min > e) min = e
    }
    return min
}*

/**
* Returns the first element\n* having the smallest value according to the provided [comparator] or ‘null’ if there are no elements.*\n*/
@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic fun ULongArray.minOrNull(): ULong? {
    if (isEmpty()) return null\n    var min = this[0]\n    for (i in 1..lastIndex) {
        val e = this[i]\n        if (min > e) min = e
    }
    return min
}*

/**
* Returns the first element\n* having the smallest value according to the provided [comparator] or ‘null’ if there are no elements.*\n*/
@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic fun UByteArray.minOrNull(): UByte? {
    if (isEmpty()) return null\n    var min = this[0]\n    for (i in 1..lastIndex) {
        val e = this[i]\n        if (min > e) min = e
    }
    return min
}*

/**
* Returns the first element\n* having the smallest value according to the provided [comparator] or ‘null’ if there are no elements.*\n*/
@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic fun UShortArray.minOrNull(): UShort? {
    if (isEmpty()) return null\n    var min = this[0]\n    for (i in 1..lastIndex) {
        val e = this[i]\n        if (min > e) min = e
    }
    return min
}*

@Deprecated("Use minWithOrNull instead.",\nReplaceWith("this.minWithOrNull(comparator)")\n)\n@DeprecatedSinceKotlin(warningSince = "1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic fun UIntArray.minWithOrNull(): UInt? {
    return minWithOrNull(comparator)
}*

@Deprecated("Use minWithOrNull instead.",\nReplaceWith("this.minWithOrNull(comparator)")\n)\n@DeprecatedSinceKotlin(warningSince = "1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic fun ULongArray.minWithOrNull(): ULong? {
    return minWithOrNull(comparator)
}*

@Deprecated("Use minWithOrNull instead.",\nReplaceWith("this.minWithOrNull(comparator)")\n)\n@DeprecatedSinceKotlin(warningSince = "1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic fun UByteArray.minWithOrNull(): UByte? {
    return minWithOrNull(comparator)
}*

@Deprecated("Use minWithOrNull instead.",\nReplaceWith("this.minWithOrNull(comparator)")\n)\n@DeprecatedSinceKotlin(warningSince = "1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic fun UShortArray.minWithOrNull(): UShort? {
    return minWithOrNull(comparator)
}*/
Comparator<in UInt>): UInt?
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}

/**
 * Returns the first element having the smallest value according to the provided [comparator] or `null` if there are no elements.
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
public fun UByteArray.minWithOrNull(comparator: Comparator<in UByte>): UByte? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}

/**
 * Returns the first element having the smallest value according to the provided [comparator] or `null` if there are no elements.
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
public fun UShortArray.minWithOrNull(comparator: Comparator<in UShort>): UShort? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}

/**
 * Returns `true` if the array has no elements.
 * 
 * @sample samples.collections.Collections.Aggregates.none
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.none(): Boolean {
    return isEmpty()
}

/**
 * Returns `true` if the array has no elements.
 * 
 * @sample samples.collections.Collections.Aggregates.none
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.none(): Boolean {
    return isEmpty()
}

/**
 * Returns `true` if the array has no elements.
 * 
 * @sample samples.collections.Collections.Aggregates.none
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.none(): Boolean {
    return isEmpty()
}

/**
 * Returns `true` if the array has no elements.
 * 
 * @sample samples.collections.Collections.Aggregates.none
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.none(): Boolean {
    return isEmpty()
}
**Open Source Used In security_automation identity-guest-cs**

```kotlin
public inline fun UByteArray.onEach(action: (UByte) -> Unit): UByteArray {
    return apply { for (element in this)
        action(element) }
}

/**
 * Performs the given [action] on each element, providing sequential index with the element,\n * and returns the array itself afterwards.\n * @param [action] function that takes the index of an element and the element itself,\n * and performs the action on the element.\n */

public inline fun UShortArray.onEach(action: (UShort) -> Unit): UShortArray {
    return apply { for (element in this)
        action(element) }
}

/**
 * Performs the given [action] on each element, providing sequential index with the element,\n * and returns the array itself afterwards.\n * @param [action] function that takes the index of an element and the element itself,\n * and performs the action on the element.\n */

public inline fun UIntArray.onEachIndexed(action: (index: Int, UInt) -> Unit): UIntArray {
    return apply {
        forEachIndexed(action) }
}

/**
 * Performs the given [action] on each element, providing sequential index with the element,\n * and returns the array itself afterwards.\n * @param [action] function that takes the index of an element and the element itself,\n * and performs the action on the element.\n */

public inline fun UByteArray.onEachIndexed(action: (index: Int, UByte) -> Unit): UByteArray {
    return apply {
        forEachIndexed(action) }
}

/**
 * Performs the given [action] on each element, providing sequential index with the element,\n * and returns the array itself afterwards.\n * @param [action] function that takes the index of an element and the element itself,\n * and performs the action on the element.\n */

public inline fun UShortArray.onEachIndexed(action: (index: Int, UShort) -> Unit): UShortArray {
    return apply {
        forEachIndexed(action) }
}

/**
 * Accumulates value starting with the first element and applying \[operation\] from left to right,\n * to current accumulator value and each element.\n * @param \[operation\] function that takes current accumulator value and an element,\n * and calculates the next accumulator value.\n * @param [operation] function that takes current accumulator value and an element,\n * and calculates the next accumulator value.\n * @param [operation] function that takes current accumulator value and an element,\n * and calculates the next accumulator value.\n */

public inline fun UIntArray.reduce(operation: (acc: UInt, UInt) -> UInt): UInt {
    if (isEmpty())
        throw
        UnsupportedOperationException("Empty array can't be reduced.\n" )
    var accumulator = this[0]
    for (index in
        1..lastIndex) {
        accumulator = operation(accumulator, this[index])\n    } return accumulator
}

/**
 * Accumulates value starting with the first element and applying \[operation\] from left to right,\n * to current accumulator value and each element.\n * @param [operation] function that takes current accumulator value and an element,\n * and calculates the next accumulator value.\n */

public inline fun ULongArray.reduce(operation: (acc: ULong, ULong) -> ULong): ULong {
    if (isEmpty())
        throw
        UnsupportedOperationException("Empty array can't be reduced.\n" )
    var accumulator = this[0]
    for (index in
        1..lastIndex) {
        accumulator = operation(accumulator, this[index])\n    } return accumulator
}

/**
 * Accumulates value starting with the first element and applying \[operation\] from left to right,\n * to current accumulator value and each element.\n * @param [operation] function that takes current accumulator value and an element,\n */

public inline fun UByteArray.reduce(operation: (acc: UByte, UByte) -> UByte): UByte {
    if (isEmpty())
        throw
        UnsupportedOperationException("Empty array can't be reduced.\n" )
    var accumulator = this[0]
    for (index in
        1..lastIndex) {
        accumulator = operation(accumulator, this[index])\n    } return accumulator
}
```

---

**Open Source Used In security_automation identity-guest-cs** 1540
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.reduce(operation: (acc: UByte, UByte) -> UByte): UByte {
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right
 * to current accumulator value and each element.\n * \n * Throws an exception if this array is empty. If the array can be empty
 * in an expected way,\n * please use [reduceOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes current accumulator value and an element,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduce
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right
 * to current accumulator value and each element with its index in the original array.\n * \n * Throws an exception if this array is empty. If the array can be empty
 * in an expected way,\n * please use [reduceIndexedOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes the index of an element, current accumulator value
 * and the element itself,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduce
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.reduceIndexed(operation: (index: Int, acc: UInt, UInt) -> UInt): UInt {
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right
 * to current accumulator value and each element with its index in the original array.\n * \n * Throws an exception if this array is empty. If the array can be empty
 * in an expected way,\n * please use [reduceIndexedOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes the index of an element, current accumulator value
 * and the element itself,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduce
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.reduceIndexed(operation: (index: Int, acc: ULong, ULong) -> ULong): ULong {
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right
 * to current accumulator value and each element with its index in the original array.\n * \n * Throws an exception if this array is empty. If the array can be empty
 * in an expected way,\n * please use [reduceIndexedOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes the index of an element, current accumulator value
 * and the element itself,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduce
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.reduceIndexed(operation: (index: Int, acc: UByte, UByte) -> UByte): UByte {
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right
 * to current accumulator value and each element with its index in the original array.\n * \n * Throws an exception if this array is empty. If the array can be empty
 * in an expected way,\n * please use [reduceIndexedOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes the index of an element, current accumulator value
 * and the element itself,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduce
 */
@sample samples.collections.Collections.Aggregates.reduce

*\{@Sample\}

```kotlin
    if (isEmpty())
        return null
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
```

Accumulates value starting from the first element and applying [operation] from left to right to current accumulator value and each element with its index in the original array. \n * \n * Returns `null` if the array is empty. \n * \n * @param [operation] function that takes the index of an element, current accumulator value and the element itself, \n * \n * and calculates the next accumulator value. \n * \n * \n```
```
```kotlin
public inline fun UIntArray.reduceIndexedOrNull(operation: (index: Int, acc: UInt, UInt) -> UInt): UInt?
    if (isEmpty())
        return null
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
```

Accumulates value starting from the first element and applying [operation] from left to right to current accumulator value and each element with its index in the original array. \n * \n * Returns `null` if the array is empty. \n * \n * @param [operation] function that takes the index of an element, current accumulator value and the element itself, \n * \n * and calculates the next accumulator value. \n * \n * \n```
```
```kotlin
public inline fun ULongArray.reduceIndexedOrNull(operation: (index: Int, acc: ULong, ULong) -> ULong): ULong?
    if (isEmpty())
        return null
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
```

Accumulates value starting from the first element and applying [operation] from left to right to current accumulator value and each element with its index in the original array. \n * \n * Returns `null` if the array is empty. \n * \n * @param [operation] function that takes the index of an element, current accumulator value and the element itself, \n * \n * and calculates the next accumulator value. \n * \n * \n```
```
```kotlin
public inline fun UByteArray.reduceIndexedOrNull(operation: (index: Int, acc: UByte, UByte) -> UByte): UByte?
    if (isEmpty())
        return null
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
```

Accumulates value starting from the first element and applying [operation] from left to right to current accumulator value and each element with its index in the original array. \n * \n * Returns `null` if the array is empty. \n * \n * @param [operation] function that takes the index of an element, current accumulator value and the element itself, \n * \n * and calculates the next accumulator value. \n * \n * \n```
```
```kotlin
public inline fun UIntArray.reduceOrNull(operation: (acc: UInt, UInt) -> UInt): UInt?
{  
    if (isEmpty())
        return null
    var accumulator = this[0]
    for (index in 1..lastIndex) {  
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
```

Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element. \n * \n * @param [operation] function that takes current accumulator value and an element, \n * \n * and calculates the next accumulator value. \n * \n * \n```
```
```kotlin
{  
    if (isEmpty())
        return null
    var accumulator = this[0]
    for (index in 1..lastIndex) {  
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
```

Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element. \n * \n * @param [operation] function that takes current accumulator value and an element, \n * \n * and calculates the next accumulator value. \n * \n * \n```
```
@kotlin.internal.InlineOnly
public inline fun ULongArray.reduceOrNull(operation: (acc: ULong, ULong) -> ULong? {  
    if (isEmpty())  
        return null
    var accumulator = this[0]
    for (index in 1..lastIndex)  
        accumulator = operation(accumulator, this[index])
    return accumulator
}  

/**
 * Accumulates value starting with the first element and applying [operation] from left to right
 * to current accumulator value and each element. Returns `null` if the array is empty.
 * [operation] function that takes current accumulator value and an element, and calculates
 * the next accumulator value.
 * 
 * @sample
 * samples.collections.Aggregates.reduceOrNull
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.reduceOrNull(operation: (acc: UByte, UByte) -> UByte? {  
    if (isEmpty())  
        return null
    var accumulator = this[0]
    for (index in 1..lastIndex)  
        accumulator = operation(accumulator, this[index])
    return accumulator
}  

/**
 * Accumulates value starting with the last element and applying [operation] from right to left
 * to current accumulator value and each element. Returns `null` if the array is empty.
 * [operation] function that takes an element and current accumulator value, and calculates
 * the next accumulator value.
 * 
 * @sample
 * samples.collections.Aggregates.reduceRight
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.reduceRight(operation: (UInt, acc: UInt) -> UInt) {  
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0)  
        accumulator = operation(get(index--), accumulator)
    return accumulator
}  

/**
 * Accumulates value starting with the last element and applying [operation] from right to left
 * to each element and current accumulator value. Returns an exception if this array is empty. If
 * the array can be empty in an expected way, please use [reduceRightOrNull] instead. It returns `null`
 * when its receiver is empty.
 * [operation] function that takes an element and current accumulator value, and calculates
 * the next accumulator value.
 * 
 * @sample
 * samples.collections.Aggregates.reduceRight
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.reduceRight(operation: (ULong, acc: ULong) -> ULong) {  
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0)  
        accumulator = operation(get(index--), accumulator)
    return accumulator
}  

/**
 * Accumulates value starting with the last element and applying [operation] from right to left
 * to each element and current accumulator value. Returns an exception if this array is empty. If
 * the array can be empty in an expected way, please use [reduceRightOrNull] instead. It returns `null`
 * when its receiver is empty.
 * [operation] function that takes an element and current accumulator value, and calculates
 * the next accumulator value.
 * 
 * @sample
 * samples.collections.Aggregates.reduceRight
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.reduceRight(operation: (UShort, acc: UShort) -> UShort) {  
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0)  
        accumulator = operation(get(index--), accumulator)
    return accumulator
}
UByteArray.reduceRight(operation: (UByte, acc: UByte) -> UByte): UByte {
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/*
 * Accumulates value starting with the last element and applying [operation] from right to
 * left to each element and current accumulator value.
 * Throws an exception if this array is empty. If the array can be empty in an expected way, use [reduceRightOrNull] instead. It returns `null` when its receiver is empty.
 * and calculates the next accumulator value.
 * function that takes an element and current accumulator value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.reduceRight(operation: (UShort, acc: UShort) -> UShort): UShort {
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/*
 * Accumulates value starting with the last element and applying [operation] from right to left to each element with its index in the original array and current accumulator value.
 * function that takes the index of an element, the element itself and current accumulator value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.reduceRightIndexed(operation: (index: Int, UInt, acc: UInt) -> UInt): UInt {
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

/*
 * Accumulates value starting with the last element and applying [operation] from right to left to each element with its index in the original array and current accumulator value.
 * function that takes the index of an element, the element itself and current accumulator value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.reduceRightIndexed(operation: (index: Int, ULong, acc: ULong) -> ULong): ULong {
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

/*
 * Accumulates value starting with the last element and applying [operation] from right to left to each element with its index in the original array and current accumulator value.
 * function that takes the index of an element, the element itself and current accumulator value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.reduceRightIndexed(operation: (index: Int, UByte, acc: UByte) -> UByte): UByte {
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

Open Source Used In security_automation identity-guest-cs 1544
* @param [operation] function that takes the index of an element, the element itself and current accumulator value,
  and calculates the next accumulator value.

@sample samples.collections.Collections.Aggregates.reduceRight

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

/**
 * Accumulates value starting with the last element and applying [operation] from right to left
 * to each element with its index in the original array and current accumulator value.
 * Returns `null` if the array is empty.
 * and calculates the next accumulator value.
 * @param [operation] function that takes the index of an element, the element itself and current accumulator value,
 * and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduceRightOrNull

@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.reduceRightIndexedOrNull(operation: (index: Int, UInt, acc: UInt) -> UInt): UInt? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

/**
 * Accumulates value starting with the last element and applying [operation] from right to left
 * to each element with its index in the original array and current accumulator value.
 * Returns `null` if the array is empty.
 * and calculates the next accumulator value.
 * @param [operation] function that takes the index of an element, the element itself and current accumulator value,
 * and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduceRightOrNull

@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.reduceRightIndexedOrNull(operation: (index: Int, ULong, acc: ULong) -> ULong): ULong? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

/**
 * Accumulates value starting with the last element and applying [operation] from right to left
 * to each element with its index in the original array and current accumulator value.
 * Returns `null` if the array is empty.
 * and calculates the next accumulator value.
 * @param [operation] function that takes the index of an element, the element itself and current accumulator value,
 * and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduceRightOrNull

@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.reduceRightIndexedOrNull(operation: (index: Int, UByte, acc: UByte) -> UByte): UByte? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

/**
 * Accumulates value starting with the last element and applying [operation] from right to left
 * to each element with its index in the original array and current accumulator value.
 * Returns `null` if the array is empty.
 * and calculates the next accumulator value.
 * @param [operation] function that takes the index of an element, the element itself and current accumulator value,
 * and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduceRightOrNull

@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

/**
 * Accumulates value starting with the last element and applying [operation] from right to left
 * to each element and current accumulator value.
 * Returns `null` if the array is empty.
 * @param [operation] function that takes an element and current accumulator value,
 * and calculates the next accumulator value.
 * @sample

@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
    var index = lastIndex
    if (index < 0) return null
    var accumulator = last
    while (index >= 0) {
        accumulator = operation(accumulator, get(index--))
        --index
    }
    return accumulator
}
@kotlin.internal.InlineOnly
public inline fun UIntArray.reduceRightOrNull(operation: (UInt, acc: UInt) -> UInt):
UInt? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with the last element and applying [operation] from right to
 * each element and current accumulator value.
 * Returns `null` if the array is empty.
 * @param [operation] function that takes an element and current accumulator value,
 * and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduceRightOrNull
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.reduceRightOrNull(operation: (ULong, acc: ULong) -> ULong):
ULong? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with the last element and applying [operation] from right to
 * each element and current accumulator value.
 * Returns `null` if the array is empty.
 * @param [operation] function that takes an element and current accumulator value,
 * and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduceRightOrNull
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.reduceRightOrNull(operation: (UByte, acc: UByte) -> UByte):
UByte? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with the last element and applying [operation] from right to
 * each element and current accumulator value.
 * Returns `null` if the array is empty.
 * @param [operation] function that takes an element and current accumulator value,
 * and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduceRightOrNull
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.reduceRightOrNull(operation: (UShort, acc: UShort) -> UShort):
UShort? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Returns a list containing successive accumulation values generated by applying
 * from left to right an `element` and current accumulator value that starts with `[initial]` value.
 * `Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect
 * the previous value in resulting list.
 * @param [operation] function that takes a current accumulator value and an
 * element, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.runningFold
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.runningFold(initial: R, operation: (acc: R, UInt) -> R): List<R> {
    if (isEmpty()) return listOf(initial)
    val result = ArrayList<R>(size + 1).apply { add(initial) }
    var accumulator = initial
    for (element in this) {
        accumulator = operation(accumulator, element)
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying
 * from left to right an `element` and current accumulator value that starts with `[initial]` value.
 * `Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect
 * the previous value in resulting list.
 * @param [operation] function that takes a current accumulator value and an
 * element, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.runningFold
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.runningFold(initial: R, operation: (acc: R, ULong) -> R): List<R> {
    if (isEmpty()) return listOf(initial)
    val result = ArrayList<R>(size + 1).apply { add(initial) }
    var accumulator = initial
    for (element in this) {
        accumulator = operation(accumulator, element)
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying
 * from left to right an `element` and current accumulator value that starts with `[initial]` value.
 * `Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect
 * the previous value in resulting list.
 * @param [operation] function that takes a current accumulator value and an
 * element, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.runningFold
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.runningFold(initial: R, operation: (acc: R, UByte) -> UByte): List<R> {
    if (isEmpty()) return listOf(initial)
    val result = ArrayList<R>(size + 1).apply { add(initial) }
    var accumulator = initial
    for (element in this) {
        accumulator = operation(accumulator, element)
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying
 * from left to right an `element` and current accumulator value that starts with `[initial]` value.
 * `Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect
 * the previous value in resulting list.
 * @param [operation] function that takes a current accumulator value and an
 * element, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.runningFold
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
    if (isEmpty()) return listOf(initial)
    val result = ArrayList<R>(size + 1).apply { add(initial) }
    var accumulator = initial
    for (element in this) {
        accumulator = operation(accumulator, element)
        result.add(accumulator)
    }
    return result
}
[operation] from left to right * to each element and current accumulator value that starts with [initial] value.

* Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.

* @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.

* @sample samples.collections.Collections.Aggregates.runningFold

```kotlin
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R>
UByteArray.runningFold(initial: R, operation: (acc: R, UByte) -> R): List<R> {
    if (isEmpty()) return listOf(initial)
    val result = ArrayList<R>(size + 1).apply { add(initial) }
    var accumulator = initial
    for (element in this) {
        accumulator = operation(accumulator, element)
        result.add(accumulator)
    }
    return result
}
```

* @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.

* @sample samples.collections.Collections.Aggregates.runningFold

```kotlin
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R>
    if (isEmpty()) return listOf(initial)
    val result = ArrayList<R>(size + 1).apply { add(initial) }
    var accumulator = initial
    for (element in this) {
        accumulator = operation(accumulator, element)
        result.add(accumulator)
    }
    return result
}
```

* @param [operation] function that takes the index of an element, current accumulator value, and the element itself, and calculates the next accumulator value.

* @sample samples.collections.Collections.Aggregates.runningFold

```kotlin
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R>
UIntArray.runningFoldIndexed(initial: R, operation: (index: Int, acc: R, UInt) -> R): List<R> {
    if (isEmpty()) return listOf(initial)
    val result = ArrayList<R>(size + 1).apply { add(initial) }
    var accumulator = initial
    for (index in indices) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}
```

* @param [operation] function that takes the index of an element, current accumulator value, and the element itself, and calculates the next accumulator value.

* @sample samples.collections.Collections.Aggregates.runningFold

```kotlin
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R>
ULongArray.runningFoldIndexed(initial: R, operation: (index: Int, acc: R, ULong) -> R): List<R> {
    if (isEmpty()) return listOf(initial)
    val result = ArrayList<R>(size + 1).apply { add(initial) }
    var accumulator = initial
    for (index in indices) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}
```

* @param [operation] function that takes the index of an element, current accumulator value, and the element itself, and calculates the next accumulator value.

* @sample samples.collections.Collections.Aggregates.runningFold

```kotlin
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R>
UByteArray.runningFoldIndexed(initial: R, operation: (index: Int, acc: R, UByte) -> R): List<R> {
    if (isEmpty()) return listOf(initial)
    val result = ArrayList<R>(size + 1).apply { add(initial) }
    var accumulator = initial
    for (index in indices) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}
```
initial\n for (index in indices) {\n accumulator = operation(index, accumulator, this[index])\n result.add(accumulator)\n }\n return result\n}/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element, its index in the original array and current accumulator value that starts with [initial] value.\n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n *\n * @param [operation] function that takes the index of an element, current accumulator value and\n * the element itself, and calculates the next accumulator value.\n *\n * @sample samples.collections.Collections.Aggregates.runningFold\n */\n@SinceKotlin("1.4\")@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R> UShortArray.runningFoldIndexed(initial: R, operation: (Int, acc: R, UShort) -> R): List<R> {\n  if (isEmpty()) return listOf(initial)\n  var result = ArrayList<R>(size + 1).apply { add(initial) }\n  var accumulator = initial\n  for (index in indices) {\n    accumulator = operation(index, accumulator, this[index])\n    result.add(accumulator)\n  }\n  return result\n}

/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element and current accumulator value that starts with the first element of this array.\n *\n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n *\n * @param [operation] function that\n * takes current accumulator value and an element, and calculates the next accumulator value.\n *\n * @sample\n */
@SinceKotlin("1.4\")@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.runningReduce(operation: (acc: UInt, UInt) -> UInt): List<UInt> {\n  if (isEmpty()) return emptyList()\n  var accumulator = this[0]\n  val result = ArrayList<UInt>(size).apply { add(accumulator) }\n  for (index in 1 until size) {\n    accumulator = operation(accumulator, this[index])\n    result.add(accumulator)\n  }\n  return result\n}

/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element and current accumulator value that starts with the first element of this array.\n *\n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n *\n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n *\n * @sample\n */
@SinceKotlin("1.4\")@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.runningReduce(operation: (acc: ULong, ULong) -> ULong): List<ULong> {\n  if (isEmpty()) return emptyList()\n  var accumulator = this[0]\n  val result = ArrayList<ULong>(size).apply { add(accumulator) }\n  for (index in 1 until size) {\n    accumulator = operation(accumulator, this[index])\n    result.add(accumulator)\n  }\n  return result\n}

/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element and current accumulator value that starts with the first element of this array.\n *\n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n *\n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n *\n * @sample\n */
@SinceKotlin("1.4\")@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.runningReduce(operation: (acc: UByte, UByte) -> UByte): List<UByte> {\n  if (isEmpty()) return emptyList()\n  var accumulator = this[0]\n  val result = ArrayList<UByte>(size).apply { add(accumulator) }\n  for (index in 1 until size) {\n    accumulator = operation(accumulator, this[index])\n    result.add(accumulator)\n  }\n  return result\n}
UShortArray.runningReduce(operation: (acc: UShort, UShort) -> UShort): List<UShort> { if (isEmpty()) return emptyList(); var accumulator = this[0]; val result = ArrayList<UShort>(size).apply { add(accumulator) } for (index in 1 until size) { accumulator = operation(accumulator, this[index]); result.add(accumulator) } return result

* Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element, its index in the original array and current accumulator value that starts with the first element of this array. Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.

@SinceKotlin("1.4") @ExperimentalUnsignedTypes @kotlin.internal.InlineOnly @public inline fun UShortArray.runningReduceIndexed(operation: (index: Int, acc: UShort, UShort) -> UShort): List<UShort> { if (isEmpty()) return emptyList(); var accumulator = this[0]; val result = ArrayList<UShort>(size).apply { add(accumulator) } for (index in 1 until size) { accumulator = operation(index, accumulator, this[index]); result.add(accumulator) } return result

* Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element, its index in the original array and current accumulator value that starts with the first element of this array. Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.

@SinceKotlin("1.4") @ExperimentalUnsignedTypes @kotlin.internal.InlineOnly @public inline fun UIntArray.runningReduceIndexed(operation: (index: Int, acc: UInt, UInt) -> UInt): List<UInt> { if (isEmpty()) return emptyList(); var accumulator = this[0]; val result = ArrayList<UInt>(size).apply { add(accumulator) } for (index in 1 until size) { accumulator = operation(index, accumulator, this[index]); result.add(accumulator) } return result

* Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element, its index in the original array and current accumulator value that starts with the first element of this array. Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.

@SinceKotlin("1.4") @ExperimentalUnsignedTypes @kotlin.internal.InlineOnly @public inline fun ULongArray.runningReduceIndexed(operation: (index: Int, acc: ULong, ULong) -> ULong): List<ULong> { if (isEmpty()) return emptyList(); var accumulator = this[0]; val result = ArrayList<ULong>(size).apply { add(accumulator) } for (index in 1 until size) { accumulator = operation(index, accumulator, this[index]); result.add(accumulator) } return result

* Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element, its index in the original array and current accumulator value that starts with the first element of this array. Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.

@SinceKotlin("1.4") @ExperimentalUnsignedTypes @kotlin.internal.InlineOnly @public inline fun UByteArray.runningReduceIndexed(operation: (index: Int, acc: UByte, UByte) -> UByte): List<UByte> { if (isEmpty()) return emptyList(); var accumulator = this[0]; val result = ArrayList<UByte>(size).apply { add(accumulator) } for (index in 1 until size) { accumulator = operation(index, accumulator, this[index]); result.add(accumulator) } return result

* Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element, its index in the original array and current accumulator value that starts with the first element of this array. Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.
starts with [initial] value.

Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.

The following code snippet uses the `scan` function to perform a cumulative aggregation on arrays:

```kotlin
public inline fun <R> UIntArray.scan(initial: R, operation: (acc: R, UInt) -> R): List<R> {
    return runningFold(initial, operation)
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element and current accumulator value that starts with [initial] value.
 * Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.
 * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.scan
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> ULongArray.scan(initial: R, operation: (acc: R, ULong) -> R): List<R> {
    return runningFold(initial, operation)
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element and current accumulator value that starts with [initial] value.
 * Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.
 * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.scan
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UByteArray.scan(initial: R, operation: (acc: R, UByte) -> R): List<R> {
    return runningFold(initial, operation)
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element and current accumulator value that starts with [initial] value.
 * Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.
 * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.scan
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
    return runningFold(initial, operation)
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element, its index in the original array and current accumulator value that starts with [initial] value.
 * Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.
 * @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.scan
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UIntArray.scanIndexed(initial: R, operation: (index: Int, acc: R, UInt) -> R): List<R> {
    return runningFoldIndexed(initial, operation)
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element, its index in the original array and current accumulator value that starts with [initial] value.
 * Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.
 * @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.scan
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> ULongArray.scanIndexed(initial: R, operation: (index: Int, acc: R, ULong) -> R): List<R> {
    return runningFoldIndexed(initial, operation)
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element, its index in the original array and current accumulator value that starts with [initial] value.
 * Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.
 * @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.scan
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UByteArray.scanIndexed(initial: R, operation: (index: Int, acc: R, UByte) -> R): List<R> {
    return runningFoldIndexed(initial, operation)
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element, its index in the original array and current accumulator value that starts with [initial] value.
 * Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.
 * @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.scan
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
    return runningFoldIndexed(initial, operation)
}
containing successive accumulation values generated by applying [operation] from left to right to each element, its index in the original array and current accumulator value that starts with [initial] value. Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list. Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list. Let's assume that the index of an element, current accumulator value, and the element itself, and calculates the next accumulator value. Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.

```kotlin
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UByteArray.scanIndexed(initial: R, operation: (index: Int, acc: R, UByte) -> R): List<R> {
    return runningFoldIndexed(initial, operation)
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element, its index in the original array and current accumulator value that starts with [initial] value. Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list. Let's assume that the index of an element, current accumulator value, and the element itself, and calculates the next accumulator value. Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.

```
@kotlin.internal.InlineOnly
@public inline fun UShortArray.scanReduceIndexed(operation: (Int, Int, UShort) -> UShort) { return runningReduceIndexed(operation) }

// Returns the sum of all values produced by [selector] function applied to each element in the array.

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun UIntArray.sumBy(selector: (UInt) -> UInt): UInt {
    var sum = 0u
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun ULongArray.sumByDouble(selector: (ULong) -> Double): Double {
    var sum = 0.0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */

@SinceKotlin("1.3")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@Suppress("INAPPLICABLE_JVM_NAME")
@kotlin.jvm.JvmName("sumOfDouble")
@ExperimentalUnsignedTypes
@public inline fun UIntArray.sumOf(selector: (UInt) -> Double): Double {
    var sum = 0.toDouble()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */

@SinceKotlin("1.3")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@Suppress("INAPPLICABLE_JVM_NAME")
@kotlin.jvm.JvmName("sumOfDouble")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun UByteArray.sumByDouble(selector: (UByte) -> Double): Double {
    var sum = 0.0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */

@SinceKotlin("1.3")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@Suppress("INAPPLICABLE_JVM_NAME")
@kotlin.jvm.JvmName("sumOfDouble")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun UShortArray.scanReduceIndexed(operation: (Int, Int, UShort) -> UShort) { return runningReduceIndexed(operation) }

// Returns the sum of all values produced by [selector] function applied to each element in the array.

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun UByteArray.sumBy(selector: (UByte) -> UInt): UInt {
    var sum = 0u
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */

@SinceKotlin("1.3")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@Suppress("INAPPLICABLE_JVM_NAME")
@kotlin.jvm.JvmName("sumOfDouble")
@ExperimentalUnsignedTypes
@public inline fun UShortArray.sumOf(selector: (UShort) -> Double): Double {
    var sum = 0.0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */

@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@Suppress("INAPPLICABLE_JVM_NAME")
@kotlin.jvm.JvmName("sumOfDouble")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun UIntArray.sumOf(selector: (UInt) -> Double): Double {
    var sum = 0.toDouble()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */

@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@Suppress("INAPPLICABLE_JVM_NAME")
@kotlin.jvm.JvmName("sumOfDouble")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun UByteArray.sumOf(selector: (UByte) -> Double): Double {
    var sum = 0.0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */

@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@Suppress("INAPPLICABLE_JVM_NAME")
@kotlin.jvm.JvmName("sumOfDouble")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun ULongArray.sumOf(selector: (ULong) -> Double): Double {
    var sum = 0.0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */

@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@Suppress("INAPPLICABLE_JVM_NAME")
@kotlin.jvm.JvmName("sumOfDouble")
@ExperimentalUnsignedTypes
@public inline fun UByteArray.sumOf(selector: (UByte) -> Double): Double {
    var sum = 0.0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */

@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@Suppress("INAPPLICABLE_JVM_NAME")
@kotlin.jvm.JvmName("sumOfDouble")
@ExperimentalUnsignedTypes
@public inline fun UShortArray.sumOf(selector: (UShort) -> Double): Double {
    var sum = 0.0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}
ByLambdaReturnType\n@Suppress("INAPPLICABLE_JVM_NAME")\n@kotlin.jvm.JvmName("sumOfDouble\n")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.sumOf(selector: (UByte) -> Double): Double {
    var sum: Double = 0.toDouble()
    for (element in this) {
        sum += selector(element)
    }
    return sum
} // * Returns the sum of all values produced by [selector] function applied to each element in the array.

*\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType\n@Suppress("INAPPLICABLE_JVM_NAME")\n@kotlin.jvm.JvmName("sumOfDouble\n")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.sumOf(selector: (UShort) -> Double): Double {
    var sum: Double = 0.toDouble()
    for (element in this) {
        sum += selector(element)
    }
    return sum
} // * Returns the sum of all values produced by [selector] function applied to each element in the array.

*\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType\n@Suppress("INAPPLICABLE_JVM_NAME")\n@kotlin.jvm.JvmName("sumOfDouble\n")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.sumOf(selector: (UInt) -> Int): Int {
    var sum: Int = 0.toInt()
    for (element in this) {
        sum += selector(element)
    }
    return sum
} // * Returns the sum of all values produced by [selector] function applied to each element in the array.

*\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType\n@Suppress("INAPPLICABLE_JVM_NAME")\n@kotlin.jvm.JvmName("sumOfDouble\n")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.sumOf(selector: (ULong) -> Int): Int {
    var sum: Int = 0.toInt()
    for (element in this) {
        sum += selector(element)
    }
    return sum
} // * Returns the sum of all values produced by [selector] function applied to each element in the array.

*\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType\n@Suppress("INAPPLICABLE_JVM_NAME")\n@kotlin.jvm.JvmName("sumOfDouble\n")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.sumOf(selector: (UByte) -> Int): Int {
    var sum: Int = 0.toInt()
    for (element in this) {
        sum += selector(element)
    }
    return sum
} // * Returns the sum of all values produced by [selector] function applied to each element in the array.

*\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType\n@Suppress("INAPPLICABLE_JVM_NAME")\n@kotlin.jvm.JvmName("sumOfDouble\n")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.sumOf(selector: (UShort) -> Int): Int {
    var sum: Int = 0.toInt()
    for (element in this) {
        sum += selector(element)
    }
    return sum
} // * Returns the sum of all values produced by [selector] function applied to each element in the array.

*\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType\n@Suppress("INAPPLICABLE_JVM_NAME")\n@kotlin.jvm.JvmName("sumOfDouble\n")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.sumOf(selector: (UInt) -> Long): Long {
    var sum: Long = 0.toLong()
    for (element in this) {
        sum += selector(element)
    }
    return sum
} // * Returns the sum of all values produced by [selector] function applied to each element in the array.

*\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType\n@Suppress("INAPPLICABLE_JVM_NAME")\n@kotlin.jvm.JvmName("sumOfDouble\n")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.sumOf(selector: (ULong) -> Long): Long {
    var sum: Long = 0.toLong()
    for (element in this) {
        sum += selector(element)
    }
    return sum
} // * Returns the sum of all values produced by [selector] function applied to each element in the array.

*\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
By Lambda ReturnType
@Suppress("INAPPLICABLE_JVM_NAME")
@kotlin.jvm.JvmName("sumOfULong")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.sumOf(selector: (UByte) -> ULong): ULong {
    var sum: ULong = 0.toULong()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */

@An@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
By Lambda ReturnType
@Suppress("INAPPLICABLE_JVM_NAME")
@kotlin.jvm.JvmName("sumOfULong")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.sumOf(selector: (UShort) -> ULong): ULong {
    var sum: ULong = 0.toULong()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.
 * The returned list has length of the shortest collection.
 */

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@public infix fun <R> UIntArray.zip(other: Array<out R>): List<Pair<UInt, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}

/**
 * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.
 * The returned list has length of the shortest collection.
 */

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@public infix fun <R> ULongArray.zip(other: Array<out R>): List<Pair<ULong, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}

/**
 * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.
 * The returned list has length of the shortest collection.
 */

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@public infix fun <R> UByteArray.zip(other: Array<out R>): List<Pair<UByte, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}

/**
 * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.
 * The returned list has length of the shortest collection.
 */

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@public infix fun <R> UShortArray.zip(other: Array<out R>): List<Pair<UShort, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}

/**
 * Returns a list of values built from the elements of `this` array and the [other] array with the same index.
 * The returned list has length of the shortest collection.
 */

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@public inline fun <R, V> UIntArray.zip(other: Array<out R>, transform: (a: UInt, b: R) -> V): List<V> {
    val size = minOf(size, other.size)
    val list = ArrayList<V>(size)
    for (i in 0 until size) {
        list.add(transform(this[i], other[i]))
    }
    return list
}

/**
 * Returns a list of values built from the elements of `this` array and the [other] array with the same index.
 * The returned list has length of the shortest collection.
 */

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@public inline fun <R, V> ULongArray.zip(other: Array<out R>, transform: (a: ULong, b: R) -> V): List<V> {
    val size = minOf(size, other.size)
    val list = ArrayList<V>(size)
    for (i in 0 until size) {
        list.add(transform(this[i], other[i]))
    }
    return list
}

/**
 * Returns a list of values built from the elements of `this` array and the [other] array with the same index.
 * The returned list has length of the shortest collection.
 */

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@public inline fun <R, V> UByteArray.zip(other: Array<out R>, transform: (a: UByte, b: R) -> V): List<V> {
    val size = minOf(size, other.size)
    val list = ArrayList<V>(size)
    for (i in 0 until size) {
        list.add(transform(this[i], other[i]))
    }
    return list
}

/**
 * Returns a list of values built from the elements of `this` array and the [other] array with the same index.
 * The returned list has length of the shortest collection.
 */

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@public inline fun <R, V> UShortArray.zip(other: Array<out R>, transform: (a: UShort, b: R) -> V): List<V> {
    val size = minOf(size, other.size)
    val list = ArrayList<V>(size)
    for (i in 0 until size) {
        list.add(transform(this[i], other[i]))
    }
    return list
}
list has length of the shortest collection.

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R, V>
UShortArray.zip(other: Array<out R>, transform: (a: UShort, b: R) -> V): List<V> { val size = minLength(size, other.size)
 val list = ArrayList<V>(size) for (i in 0 until size) { list.add(transform(this[i], other[i])) } return list } // Returns a list of pairs built from the elements of `this` collection and [other] array with the same index. * The returned list has length of the shortest collection.

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R, V>
UIntArray.zip(other: Iterable<R>, transform: (a: UInt, b: R) -> V): List<V> { val size = this.size
 var listSize = size
 val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), listSize)) var i = 0
 for (element in other) { if (i >= listSize) break
 list.add(transform(this[i++], element)) } return list } // Returns a list of values built from the elements of `this` array and the [other] collection with the same index in * using the provided [transform] function applied to each pair of elements. * The returned list has length of the shortest collection.

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R, V>
UByteArray.zip(other: Iterable<R>, transform: (a: UByte, b: R) -> V): List<V> { val size = this.size
 var listSize = size
 val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), listSize)) var i = 0
 for (element in other) { if (i >= listSize) break
 list.add(transform(this[i++], element)) } return list } // Returns a list of values built from the elements of `this` array and the [other] collection with the same index in * using the provided [transform] function applied to each pair of elements. * The returned list has length of the shortest collection.

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R, V>
ULongArray.zip(other: Iterable<R>, transform: (a: ULong, b: R) -> V): List<V> { val size = this.size
 var listSize = size
 val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), listSize)) var i = 0
 for (element in other) { if (i >= listSize) break
 list.add(transform(this[i++], element)) } return list } // Returns a list of values built from the elements of `this` array and the [other] collection with the same index in * using the provided [transform] function applied to each pair of elements. * The returned list has length of the shortest collection.
```

*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly@public inline fun <R, V>
UShortArray.zip(other: Array<out R>, transform: (a: UShort, b: R) -> V): List<V> { val size = minLength(size, other.size)
 val list = ArrayList<V>(size) for (i in 0 until size) { list.add(transform(this[i], other[i])) }
 return list } // Returns a list of pairs built from the elements of `this` collection and [other] array with the same index. * The returned list has length of the shortest collection.

*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly@public inline fun <R, V>
UIntArray.zip(other: Iterable<R>, transform: (a: UInt, b: R) -> V): List<V> { val size = this.size
 var listSize = size
 val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), listSize)) var i = 0
 for (element in other) { if (i >= listSize) break
 list.add(transform(this[i++], element)) } return list } // Returns a list of values built from the elements of `this` array and the [other] collection with the same index in * using the provided [transform] function applied to each pair of elements. * The returned list has length of the shortest collection.

*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly@public inline fun <R, V>
UByteArray.zip(other: Iterable<R>, transform: (a: UByte, b: R) -> V): List<V> { val size = this.size
 var listSize = size
 val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), listSize)) var i = 0
 for (element in other) { if (i >= listSize) break
 list.add(transform(this[i++], element)) } return list } // Returns a list of values built from the elements of `this` array and the [other] collection with the same index in * using the provided [transform] function applied to each pair of elements. * The returned list has length of the shortest collection.

*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly@public inline fun <R, V>
ULongArray.zip(other: Iterable<R>, transform: (a: ULong, b: R) -> V): List<V> { val size = this.size
 var listSize = size
 val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), listSize)) var i = 0
 for (element in other) { if (i >= listSize) break
 list.add(transform(this[i++], element)) } return list } // Returns a list of values built from the elements of `this` array and the [other] collection with the same index in * using the provided [transform] function applied to each pair of elements. * The returned list has length of the shortest collection.

*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly@public inline fun <R, V>
UShortArray.zip(other: Iterable<R>, transform: (a: UShort, b: R) -> V): List<V> { val size = this.size
 var listSize = size
 val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), listSize)) var i = 0
 for (element in other) { if (i >= listSize) break
 list.add(transform(this[i++], element)) } return list } // Returns a list of values built from the elements of `this` array and the [other] collection with the same index in * using the provided [transform] function applied to each pair of elements. * The returned list has length of the shortest collection.

list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))
    var i = 0
    for (element in other) {
        if (i >= arraySize) break
        list.add(transform(this[i++], element))
    }
    return list
}

/**
 * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.
 * The returned list has length of the shortest collection.
 * @sample samples.collections.Iterables.zip
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public infix fun UByteArray.zip(other: UByteArray): List<Pair<UByte, UByte>> {
    val size = minOf(size, other.size)
    val list = ArrayList<Pair<UByte, UByte>>(size)
    for (i in 0 until size) {
        list.add(transform(this[i], other[i]))
    }
    return list
}

/**
 * Returns the sum of all elements in the array.
 */
@kotlin.jvm.JvmName("sumOfUInt")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun
Array<out UInt>.sum(): UInt |\n    var sum: UInt = 0u\n    for (element in this) |\n        sum += element\n    return sum\n|\n
*\n@kotlin.jvm.JvmName("sumOfUInt")\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n\n\n\npublic fun Array<out ULong>.sum(): ULong |\n    var sum: ULong = 0uL\n    for (element in this) |\n        sum += element\n    return sum\n|\n
*\n@kotlin.jvm.JvmName("sumOfULong")\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n\n\n\npublic fun Array<out UByte>.sum(): UInt |\n    var sum: UInt = 0u\n    for (element in this) |\n        sum += element\n    return sum\n|\n
*\n@kotlin.jvm.JvmName("sumOfUByte")\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n\n\n\npublic fun Array<out UShort>.sum(): UInt |\n    var sum: UInt = 0u\n    for (element in this) |\n        sum += element\n    return sum\n|\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n\n\n\npublic fun UIntArray.sum(): UInt |\n    return storage.sum().toUInt()\n|\n
*\n@file:kotlin.jvm.JvmMultifileClass\n@file:kotlin.jvm.JvmName("UCollectionsKt")\npackage kotlin.collections\n\nNOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt file:// See: https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib\n
import kotlin.random.*
import kotlin.ranges.contains
import kotlin.ranges.reversed

/**
 * Returns an array of UByte containing all of the
 * elements of this collection.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes\n\n\n\npublic fun Collection<UByte>.toUByteArray(): UByteArray |\n    val result = UByteArray(size)\n    var index = 0\n    for (element in this)
        result[index++] = element\n    return result\n|\n
/**
 * Returns an array of UInt containing all of the
 * elements of this collection.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes\n\n\n\npublic fun Collection<UInt>.toUIntArray(): UIntArray |\n    val result = UIntArray(size)\n    var index = 0\n    for (element in this)
        result[index++] = element\n    return result\n|\n
/**
 * Returns an array of ULong containing all of the
 * elements of this collection.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes\n\n\n\npublic fun Collection<ULong>.toULongArray(): ULongArray |\n    val result = ULongArray(size)\n    var index = 0\n    for (element in this)
        result[index++] = element\n    return result\n|\n
/**
 * Returns an array of UShort containing all of the
 * elements of this collection.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes\n\n\n\npublic fun Collection<UShort>.toUShortArray(): UShortArray |\n    val result = UShortArray(size)\n    var index = 0\n    for (element in this)
        result[index++] = element\n    return result\n|
Iterables<UInt>.sum(): UInt {\n  var sum: UInt = 0u\n  for (element in this) {\n    sum += element\n  }\n  return sum\n}\n/* * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n * See: https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib\n */\n
Iterable<UShort>.sum(): UInt {
    var sum: UInt = 0u
    for (element in this) {
        sum += element
    }
    return sum
}

/*
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("UComparisonsKt")

package kotlin.comparisons

//
// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt
// See: https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib
//

import kotlin.random.*

/**
 * Returns the greater of two values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun maxOf(a: UInt, b: UInt): UInt {
    return if (a >= b) a else b
}

/**
 * Returns the greater of two values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun maxOf(a: ULong, b: ULong): ULong {
    return if (a >= b) a else b
}

/**
 * Returns the greater of two values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun maxOf(a: UByte, b: UByte): UByte {
    return if (a >= b) a else b
}

/**
 * Returns the greater of two values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun maxOf(a: UShort, b: UShort): UShort {
    return if (a >= b) a else b
}

/**
 * Returns the greater of three values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun maxOf(a: UInt, b: UInt, c: UInt): UInt {
    return maxOf(maxOf(a, b), c)
}

/**
 * Returns the greater of three values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun maxOf(a: ULong, b: ULong, c: ULong): ULong {
    return maxOf(maxOf(a, b), c)
}

/**
 * Returns the greater of three values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun maxOf(a: UByte, b: UByte, c: UByte): UByte {
    return maxOf(maxOf(a, b), c)
}

/**
 * Returns the greater of three values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
    return maxOf(maxOf(a, b), c)
}

/**
 * Returns the smaller of two values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun minOf(a: UInt, b: UInt): UInt {
    return if (a <= b) a else b
}

/**
 * Returns the smaller of two values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun minOf(a: ULong, b: ULong): ULong {
    return if (a <= b) a else b
}

/**
 * Returns the smaller of two values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun minOf(a: UByte, b: UByte): UByte {
    return if (a <= b) a else b
}

/**
 * Returns the smaller of two values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun minOf(a: UShort, b: UShort): UShort {
    return if (a <= b) a else b
}

/**
 * Returns the smaller of three values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun minOf(a: UInt, b: UInt, c: UInt): UInt {
    return minOf(minOf(a, b), c)
}

/**
 * Returns the smaller of three values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun minOf(a: ULong, b: ULong, c: ULong): ULong {
    return minOf(minOf(a, b), c)
}

/**
 * Returns the smaller of three values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun minOf(a: UByte, b: UByte, c: UByte): UByte {
    return minOf(minOf(a, b), c)
}

/**
 * Returns the smaller of three values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
    return minOf(minOf(a, b), c)
}

Open Source Used In security_automation identity-guest-cs 1559
of three values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun minOf(a: UShort, b: UShort, c: UShort): UShort {
    return minOf(a, minOf(b, c))
}

/**
 * Returns the smaller of the given values.
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@public fun minOf(a: UInt, vararg other: UInt): UInt {
    var min = a
    for (e in other) min = minOf(min, e)
    return min
}

/**
 * Returns the smaller of the given values.
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@public fun minOf(a: ULong, vararg other: ULong): ULong {
    var min = a
    for (e in other) min = minOf(min, e)
    return min
}

/**
 * Returns the smaller of the given values.
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@public fun minOf(a: UByte, vararg other: UByte): UByte {
    var min = a
    for (e in other) min = minOf(min, e)
    return min
}

/**
 * Returns the smaller of the given values.
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@public fun minOf(a: UShort, vararg other: UShort): UShort {
    var min = a
    for (e in other) min = minOf(min, e)
    return min
}

"*/

@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("URangesKt")

package kotlin.ranges

//
// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt
// See: https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib

import kotlin.random.*

/**
 * Returns a random element from this range.
 * 
 * @throws IllegalArgumentException if this range is empty.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun UIntRange.random(): UInt {
    return random(Random)
}

/**
 * Returns a random element from this range.
 * 
 * @throws IllegalArgumentException if this range is empty.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun ULongRange.random(): ULong {
    return random(Random)
}

/**
 * Returns a random element from this range using the specified source of randomness.
 * 
 * @throws IllegalArgumentException if this range is empty.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@public fun UIntRange.random(random: Random): UInt {
    try {
        return random.nextUInt(this)
    } catch(e: IllegalArgumentException) {
        throw NoSuchElementException(e.message)
    }
}

/**
 * Returns a random element from this range using the specified source of randomness.
 * 
 * @throws IllegalArgumentException if this range is empty.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@public fun ULongRange.random(random: Random): ULong {
    try {
        return random.nextULong(this)
    } catch(e: IllegalArgumentException) {
        throw NoSuchElementException(e.message)
    }
}

/**
 * Returns a random element from this range, or `null` if this range is empty.
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun UIntRange.randomOrNull(): UInt? {
    return randomOrNull(Random)
}

/**
 * Returns a random element from this range, or `null` if this range is empty.
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun ULongRange.randomOrNull(): ULong? {
    return randomOrNull(Random)
}

/**
 * Returns `true` if this range contains the specified `element`.
 * 
 * Always returns `false` if the `element` is `null`.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline operator fun UIntRange.contains(element: UInt?): Boolean {
    return element != null && contains(element)
}

@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public fun UIntRange.randomOrNull(random: Random): UInt? {
    if (isEmpty())
        return null
    return random.nextUInt(this)
}

@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public fun ULongRange.randomOrNull(random: Random): ULong? {
    if (isEmpty())
        return null
    return random.nextULong(this)
}

@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@public fun UIntRange.randomOrNull(random: Random): UInt? {
    if (isEmpty())
        return null
    return random.nextUInt(this)
}

@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@public fun ULongRange.randomOrNull(random: Random): ULong? {
    if (isEmpty())
        return null
    return random.nextULong(this)
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline operator fun UIntRange.contains(element: UInt?): Boolean {
    return element != null && contains(element)
}
Returns `true` if this range contains the specified `value`.

```kotlin
public operator fun ULongRange.contains(value: ULong?): Boolean {
    return value != null &&
    contains(value.toULong())
}
```

```kotlin
/**
 * Checks if the specified `value` belongs to this range.
 */
public operator fun ULongRange.contains(value: UByte): Boolean {
    return contains(value.toULong())
}
```

```kotlin
/**
 * Checks if the specified `value` belongs to this range.
 */
public operator fun ULongRange.contains(value: UInt): Boolean {
    return contains(value.toULong())
}
```

```kotlin
/**
 * Checks if the specified `value` belongs to this range.
 */
public operator fun ULongRange.contains(value: UShort): Boolean {
    return contains(value.toULong())
}
```

```kotlin
/**
 * Returns a progression from this value down to the specified `to` value with the step -1.
 * The `to` value should be less than or equal to `this` value. If the `to` value is greater than `this` value the returned progression is empty.
 */
public infix fun UByte.downTo(to: UByte): UIntProgression {
    return UIntProgression.fromClosedRange(this.toUInt(), to.toUInt(), -1)
}
```

```kotlin
/**
 * Returns a progression from this value down to the specified `to` value with the step -1.
 * The `to` value should be less than or equal to `this` value. If the `to` value is greater than `this` value the returned progression is empty.
 */
public infix fun UInt.downTo(to: UInt): UIntProgression {
    return UIntProgression.fromClosedRange(this, to, -1)
}
```

```kotlin
/**
 * Returns a progression that goes over the same range in the opposite direction with the step -1.
 */
public fun UIntProgression.reversed(): UIntProgression {
    return UIntProgression.fromClosedRange(last, first, -step)
}
```

```kotlin
/**
 * Returns a progression that goes over the same range with the given step.
 */
public infix fun ULongProgression.step(step: Long): ULongProgression {
    return ULongProgression.fromClosedRange(last, first, step)
}
```

```kotlin
/**
 * Returns a progression that goes over the same range with the given step.
 */
public infix fun ULongProgression.step(step: Int): UIntProgression {
    return UIntProgression.fromClosedRange(last, first, step)
}
```

```kotlin
/**
 * Returns a progression that goes over the same range with the given step.
 */
public fun ULongProgression.fromClosedRange(first: ULong, last: ULong): ULongProgression {
    return ULongProgression.fromClosedRange(first, last)
}
```
then the returned range is empty.

@ExperimentalUnsignedTypes
@public infix fun UByte.until(to: UByte): UIntRange {
    if (to <= UByte.MIN_VALUE) return UIntRange.EMPTY
    return this.toUInt() .. (to - 1u)
}

@ExperimentalUnsignedTypes
@public infix fun UInt.until(to: UInt): UIntRange {
    if (to <= UInt.MIN_VALUE) return UIntRange.EMPTY
    return this .. (to - 1u)
}

@ExperimentalUnsignedTypes
@public infix fun ULong.until(to: ULong): ULongRange {
    if (to <= ULong.MIN_VALUE) return ULongRange.EMPTY
    return toULong() .. (to - 1u)
}

@ExperimentalUnsignedTypes
@public infix fun UShort.until(to: UShort): UIntRange {
    if (to <= UShort.MIN_VALUE) return UIntRange.EMPTY
    return this.toUInt() .. (to - 1u)
}

@ExperimentalUnsignedTypes
@public fun UInt.coerceAtLeast(minimumValue: UInt): UInt {
    return if (this < minimumValue) minimumValue else this
}

@ExperimentalUnsignedTypes
@public fun ULong.coerceAtLeast(minimumValue: ULong): ULong {
    return if (this < minimumValue) minimumValue else this
}

@ExperimentalUnsignedTypes
@public fun UByte.coerceAtLeast(minimumValue: UByte): UByte {
    return if (this < minimumValue) minimumValue else this
}

@ExperimentalUnsignedTypes
@public fun UShort.coerceAtLeast(minimumValue: UShort): UShort {
    return if (this < minimumValue) minimumValue else this
}

@ExperimentalUnsignedTypes
@public fun UInt.coerceAtMost(maximumValue: UInt): UInt {
    return if (this > maximumValue) maximumValue else this
}

@ExperimentalUnsignedTypes
@public fun ULong.coerceAtMost(maximumValue: ULong): ULong {
    return if (this > maximumValue) maximumValue else this
}

@ExperimentalUnsignedTypes
@public fun UByte.coerceAtMost(maximumValue: UByte): UByte {
    return if (this > maximumValue) maximumValue else this
}

@ExperimentalUnsignedTypes
@public fun UShort.coerceAtMost(maximumValue: UShort): UShort {
    return if (this > maximumValue) maximumValue else this
}
value is not greater than the specified [maximumValue].

* @return this value if it's less than or equal to the [maximumValue] or the [maximumValue] otherwise.

* @sample samples.comparisons.ComparableOps.coerceAtMostUnsigned

*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UShort.coerceAtMost(maximumValue: UShort): UShort {\n    return if (this > maximumValue) maximumValue else this\n}\n
*\nEnsures that this value lies in the specified range [minimumValue]..[maximumValue].

* @return this value if it's in the range, or [minimumValue] if this value is less than [minimumValue], or [maximumValue] if this value is greater than [maximumValue].

* @sample samples.comparisons.ComparableOps.coerceAtMostUnsigned

*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UInt.coerceIn(minimumValue: UInt, maximumValue: UInt): UInt {\n    if (minimumValue > maximumValue) throw IllegalArgumentException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")\n    return this\n}\n
*\nEnsures that this value lies in the specified range [minimumValue]..[maximumValue].

* @return this value if it's in the range, or [minimumValue] if this value is less than [minimumValue], or [maximumValue] if this value is greater than [maximumValue].

* @sample samples.comparisons.ComparableOps.coerceInUnsigned

*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun ULong.coerceIn(minimumValue: ULong, maximumValue: ULong): ULong {\n    if (minimumValue > maximumValue) throw IllegalArgumentException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")\n    return this\n}\n
*\nEnsures that this value lies in the specified range [minimumValue]..[maximumValue].

* @return this value if it's in the range, or [minimumValue] if this value is less than [minimumValue], or [maximumValue] if this value is greater than [maximumValue].

* @sample samples.comparisons.ComparableOps.coerceInUnsigned

*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UByte.coerceIn(minimumValue: UByte, maximumValue: UByte): UByte {\n    if (minimumValue > maximumValue) throw IllegalArgumentException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")\n    return this\n}\n
*\nEnsures that this value lies in the specified range [minimumValue]..[maximumValue].

* @return this value if it's in the range, or [minimumValue] if this value is less than [minimumValue], or [maximumValue] if this value is greater than [maximumValue].

* @sample samples.comparisons.ComparableOps.coerceInUnsigned

*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UShort.coerceIn(minimumValue: UShort, maximumValue: UShort): UShort {\n    if (minimumValue > maximumValue) throw IllegalArgumentException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")\n    return this\n}\n
*\nEnsures that this value lies in the specified range [minimumValue]..[maximumValue].

* @return this value if it's in the range, or [minimumValue] if this value is less than [minimumValue], or [maximumValue] if this value is greater than [maximumValue].

* @sample samples.comparisons.ComparableOps.coerceInUnsigned

*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UInt.coerceIn(range: ClosedRange<UInt>): UInt {\n    if (range is ClosedFloatingPointRange) {\n        return this.coerceIn<UInt>(range)\n    } else if (range.isEmpty()) throw IllegalArgumentException("Cannot coerce value to an empty range: $range.")\n    return when {\n        this < range.start -> range.start\n        this > range.endInclusive -> range.endInclusive\n        else -> this\n    }\n}\n
*\nEnsures that this value lies in the specified [range].

* @return this value if it's in the [range], or `range.start` if this value is less than `range.start`, or `range.endInclusive` if this value is greater than `range.endInclusive`. 

* @sample samples.comparisons.ComparableOps.coerceInUnsigned

*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun ULong.coerceIn(range: ClosedRange<UInt>): ULong {\n    \n}\n
*\nEnsures that this value lies in the specified [range].

* @return this value if it's in the [range], or `range.start` if this value is less than `range.start`, or `range.endInclusive` if this value is greater than `range.endInclusive`. 

* @sample samples.comparisons.ComparableOps.coerceInUnsigned

*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UByte.coerceIn(range: ClosedRange<UInt>): UByte {\n    \n}\n
*\nEnsures that this value lies in the specified [range].

* @return this value if it's in the [range], or `range.start` if this value is less than `range.start`, or `range.endInclusive` if this value is greater than `range.endInclusive`. 

* @sample samples.comparisons.ComparableOps.coerceInUnsigned
ClosedRange<ULong>): ULong {
    if (range is ClosedFloatingPointRange) {
        return this.coerceIn<ULong>(range)
    }
    if (range.isEmpty()) throw IllegalArgumentException("Cannot coerce value to an empty range: $range.

    return when {
        this < range.start -> range.start
        this > range.endInclusive -> range.endInclusive
        else -> this
    }
}

/*
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("USequencesKt")

package kotlin.sequences

import kotlin.random.*

/**
 * Returns the sum of all elements in the sequence.
 *
 * The operation is _terminal_.
 */
@kotlin.jvm.JvmName("sumOfUInt")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Sequence<UInt>.sum(): UInt {
    var sum: UInt = 0u
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the sequence.
 *
 * The operation is _terminal_.
 */
@kotlin.jvm.JvmName("sumOfULong")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Sequence<ULong>.sum(): ULong {
    var sum: ULong = 0uL
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the sequence.
 *
 * The operation is _terminal_.
 */
@kotlin.jvm.JvmName("sumOfUByte")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Sequence<UByte>.sum(): UInt {
    var sum: UInt = 0u
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the sequence.
 *
 * The operation is _terminal_.
 */
@kotlin.jvm.JvmName("sumOfUShort")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Sequence<UShort>.sum(): UInt {
    var sum: UInt = 0u
    for (element in this) {
        sum += element
    }
    return sum
}

/*
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin

public expect open class Error : Throwable {
    constructor()
    constructor(message: String?)
    constructor(message: String?, cause: Throwable?)
    constructor(cause: Throwable?)
}

public expect open class Exception : Throwable {
    constructor()
    constructor(message: String?)
    constructor(message: String?, cause: Throwable?)
    constructor(cause: Throwable?)
}

public expect open class RuntimeException : Exception {
    constructor()
    constructor(message: String?)
    constructor(message: String?, cause: Throwable?)
    constructor(cause: Throwable?)
}

public expect open class IllegalArgumentException : RuntimeException {
    constructor()
    constructor(message: String?)
    constructor(message: String?, cause: Throwable?)
    constructor(cause: Throwable?)
}

public expect open class IllegalStateException : RuntimeException {
    constructor()
    constructor(message: String?)
    constructor(message: String?, cause: Throwable?)
    constructor(cause: Throwable?)
}

public expect open class IndexOutOfBoundsException : RuntimeException {
    constructor()
    constructor(message:
    String?)
    constructor(message: String?, cause: Throwable?)
    constructor(cause: Throwable?)
}

public expect open class UnsupportedOperationException : RuntimeException {
    constructor()
    constructor(message: String?)
    constructor(message: String?, cause: Throwable?)
    constructor(cause: Throwable?)
}

public expect open class NumberFormatException : IllegalArgumentException {
    constructor()
    constructor(message: String?)
}

public expect open class NullPointerException : RuntimeException {
    constructor()
    constructor(message: String?)
}

public expect open class ClassCastException : RuntimeException {
    constructor()
    constructor(message: String?)
}

public expect open class AssertionError : Error {
    constructor()
    constructor(message: Any?)
}

public expect open class NoSuchElementException : RuntimeException {
    constructor()
    constructor(message: String?)
}

public expect open class ConcurrentModificationException : RuntimeException {
    constructor()
    constructor(message: String?)
}

public expect open class UnsupportedOperationException : RuntimeException {
    constructor()
    constructor(message: String?)
}

public expect open class NumberFormatException : IllegalArgumentException {
    constructor()
    constructor(message: String?)
}

public expect open class NullPointerException : RuntimeException {
    constructor()
    constructor(message: String?)
}

public expect open class ClassCastException : RuntimeException {
    constructor()
    constructor(message: String?)
}

public expect open class AssertionError : Error {
    constructor()
    constructor(message: Any?)
}

public expect open class NoSuchElementException : RuntimeException {
    constructor()
    constructor(message: String?)
}

public expect open class ConcurrentModificationException : RuntimeException {
    constructor()
    constructor(message: String?)
}

public expect open class UnsupportedOperationException : RuntimeException {
    constructor()
    constructor(message: String?)
}

public expect open class NumberFormatException : IllegalArgumentException {
    constructor()
    constructor(message: String?)
}

public expect open class NullPointerException : RuntimeException {
    constructor()
    constructor(message: String?)
}

public expect open class ClassCastException : RuntimeException {
    constructor()
    constructor(message: String?)
}

public expect open class AssertionError : Error {
    constructor()
    constructor(message: Any?)
}

public expect open class NoSuchElementException : RuntimeException {
    constructor()
    constructor(message: String?)
}

public expect open class ConcurrentModificationException : RuntimeException {
    constructor()
    constructor(message: String?)
}

public expect open class UnsupportedOperationException : RuntimeException {
    constructor()
    constructor(message: String?)
}

public expect open class NumberFormatException : IllegalArgumentException {
    constructor()
    constructor(message: String?)
}

public expect open class NullPointerException : RuntimeException {
    constructor()
    constructor(message: String?)
}

public expect open class ClassCastException : RuntimeException {
    constructor()
    constructor(message: String?)
}

public expect open class AssertionError : Error {
    constructor()
    constructor(message: Any?)
}

public expect open class NoSuchElementException : RuntimeException {
    constructor()
    constructor(message: String?)
}

public expect open class ConcurrentModificationException : RuntimeException {
    constructor()
    constructor(message: String?)
}

public expect open class UnsupportedOperationException : RuntimeException {
    constructor()
    constructor(message: String?)
}

public expect open class NumberFormatException : IllegalArgumentException {
    constructor()
    constructor(message: String?)
}

public expect open class NullPointerException : RuntimeException {
    constructor()
    constructor(message: String?)
}

public expect open class ClassCastException : RuntimeException {
    constructor()
    constructor(message: String?)
}

public expect open class AssertionError : Error {
    constructor()
    constructor(message: Any?)
}

public expect open class NoSuchElementException : RuntimeException {
    constructor()
    constructor(message: String?)
}

public expect open class ConcurrentModificationException : RuntimeException {
    constructor()
    constructor(message: String?)
}

public expect open class UnsupportedOperationException : RuntimeException {
    constructor()
    constructor(message: String?)
}

public expect open class NumberFormatException : IllegalArgumentException {
    constructor()
    constructor(message: String?)
}

public expect open class NullPointerException : RuntimeException {
    constructor()
    constructor(message: String?)
}

public expect open class ClassCastException : RuntimeException {
    constructor()
    constructor(message: String?)
}

public expect open class AssertionError : Error {
    constructor()
    constructor(message: Any?)
}

public expect open class NoSuchElementException : RuntimeException {
    constructor()
    constructor(message: String?)
}

public expect open class ConcurrentModificationException : RuntimeException {
    constructor()
    constructor(message: String?)
}

public expect open class UnsupportedOperationException : RuntimeException {
    constructor()
    constructor(message: String?)
}

public expect open class NumberFormatException : IllegalArgumentException {
    constructor()
    constructor(message: String?)
}

public expect open class NullPointerException : RuntimeException {
    constructor()
    constructor(message: String?)
}

public expect open class ClassCastException : RuntimeException {
    constructor()
    constructor(message: String?)
}

public expect open class AssertionError : Error {
    constructor()
    constructor(message: Any?)
}

public expect open class NoSuchElementException : RuntimeException {
    constructor()
    constructor(message: String?)
}

public expect open class ConcurrentModificationException : RuntimeException {
    constructor()
    constructor(message: String?)
}

public expect open class UnsupportedOperationException : RuntimeException {
    constructor()
    constructor(message: String?)
}

public expect open class NumberFormatException : IllegalArgumentException {
    constructor()
    constructor(message: String?)
}

public expect open class NullPointerException : RuntimeException {
    constructor()
    constructor(message: String?)
}

public expect open class ClassCastException : RuntimeException {
    constructor()
    constructor(message: String?)
}

public expect open class AssertionError : Error {
    constructor()
    constructor(message: Any?)
}

public expect open class NoSuchElementException : RuntimeException {
    constructor()
    constructor(message: String?)
}

public expect open class ConcurrentModificationException : RuntimeException {
    constructor()
    constructor(message: String?)
}

public expect open class UnsupportedOperationException : RuntimeException {
    constructor()
    constructor(message: String?)
}

public expect open class NumberFormatException : IllegalArgumentException {
    constructor()
    constructor(message: String?)
}

public expect open class NullPointerException : RuntimeException {
    constructor()
    constructor(message: String?)
}

public expect open class ClassCastException : RuntimeException {
    constructor()
    constructor(message: String?)
}

public expect open class AssertionError : Error {
    constructor()
    constructor(message: Any?)
}

public expect open class NoSuchElementException : RuntimeException {
    constructor()
    constructor(message: String?)
}

public expect open class ConcurrentModificationException : RuntimeException {
    constructor()
    constructor(message: String?)
}

public expect open class UnsupportedOperationException : RuntimeException {
    constructor()
    constructor(message: String?)
}

public expect open class NumberFormatException : IllegalArgumentException {
    constructor()
    constructor(message: String?)
}

public expect open class NullPointerException : RuntimeException {
    constructor()
    constructor(message: String?)
}

public expect open class ClassCastException : RuntimeException {
    constructor()
    constructor(message: String?)
}

public expect open class AssertionError : Error {
    constructor()
    constructor(message: Any?)
}

public expect open class NoSuchElementException : RuntimeException {
    constructor()
    constructor(message: String?)
}

public expect open class ConcurrentModificationException : RuntimeException {
    constructor()
    constructor(message: String?)
}

public expect open class UnsupportedOperationException : RuntimeException {
    constructor()
    constructor(message: String?)
}

public expect open class NumberFormatException : IllegalArgumentException {
    constructor()
    constructor(message: String?)
}

public expect open class NullPointerException : RuntimeException {
    constructor()
    constructor(message: String?)
}

public expect open class ClassCastException : RuntimeException {
    constructor()
    constructor(message: String?)
}

public expect open class AssertionError : Error {
    constructor()
    constructor(message: Any?)
}

public expect open class NoSuchElementException : RuntimeException {
    constructor()
    constructor(message: String?)
}

public expect open class ConcurrentModificationException : RuntimeException {
    constructor()
    constructor(message: String?)
}

public expect open class UnsupportedOperationException : RuntimeException {
    constructor()
    constructor(message: String?)
}

public expect open class NumberFormatException : IllegalArgumentException {
    constructor()
    constructor(message: String?)
}

public expect open class NullPointerExce...
Open Source Used in security_automation identity-guest-cs 1565

```kotlin
package kotlin.js

import kotlin.annotation.AnnotationTarget.*

/**
 * Gives a declaration (a function, a property or a class) specific name in JavaScript.
 */
@Target(CLASS, FUNCTION, PROPERTY, CONSTRUCTOR, PROPERTY_GETTER, PROPERTY_SETTER)
@OptionalExpectation
public expect annotation class JsName(val name: String)

/**
 * Marks experimental JS export annotations.
 *
 * Note that behavior of these annotations will likely be changed in the future.
 * Usages of such annotations will be reported as warnings unless an explicit opt-in with
 * the `[OptIn]` annotation, e.g. `@OptIn(ExperimentalJsExport::class)`, or with the `-Xopt-in=kotlin.js.ExperimentalJsExport` compiler option
 * is given.
 */
@OptIn(ExperimentalJsExport::class)
@Suppress("DEPRECATION")
@RequiresOptIn level = RequiresOptIn.Level.WARN
@RequiresOptIn(level = RequiresOptIn.Level.WARNING)
@MustBeDocumented
@Retention(AnnotationRetention.BINARY)
public expect annotation class ExperimentalJsExport
```

---

`Open Source Used in security_automation identity-guest-cs 1565`
the type of map keys. The map is invariant in its key type, as it can accept key as a parameter (of [containsKey] for example) and return it in [keys] set. */

public interface Map<K, V> {
    // Query Operations
    /**
     * Returns the number of key/value pairs in the map.
     * @param V the type of map values. The map is covariant in its value type.
     * @param V the type of map values. The map is covariant in its value type.
     */
    public int size();
    // Modification Operations
    public V getOrDefault(K key, V defaultValue);
    public V get(K key);
    public void put(K key, V value);
    public V remove(K key);
    public V putIfAbsent(K key, V value);
    public V replace(K key, V oldValue, V newValue);
    public V replaceIfPresent(K key, V oldValue, V newValue);
    public V compute(K key, BiFunction<? super K, ? super V, ? super V> remapper);
    public V computeIfAbsent(K key, BiFunction<? super K, ? super V, V> remapper);
    public V computeIfPresent(K key, BiFunction<? super K, ? super V, V> remapper);
    public boolean remove(K key);
    public boolean containsKey(K key);
    public boolean containsValue(V value);
    public boolean isEmpty();
    public boolean isPartialOrder();
    public int cardinality();
    public int ceilingValue(K key);
    public int ceilingKey(V value);
    public int floorValue(K key);
    public int floorKey(V value);
    public int firstKey();
    public int lastKey();
    public int rank(K key);
    public int rank(V value);
    public int lowerKey(V value);
    public int higherKey(V value);
    public int lowerEntry(K key);
    public int higherEntry(K key);
    public int floorEntry(K key);
    public int ceilingEntry(K key);
    public int lowerEntry(V value);
    public int higherEntry(V value);
    public int floorEntry(V value);
    public int ceilingEntry(V value);
    public int entrySet();
    public Collection<V> values();
    public Collection<K> keys();
    public Map<K, V> subMap(K fromKey, boolean fromInclusive);
    public Map<K, V> subMap(K fromKey, K toKey, boolean fromInclusive, boolean toInclusive);
    public Map<K, V> headMap(K toKey, boolean toInclusive);
    public Map<K, V> tailMap(K fromKey, boolean fromInclusive);
    public Map<K, V> entrySet();
    public Set<Map.Entry<K, V>> entrySet();
    public Set<Map.Entry<K, V>> keySet();
    public Set<Map.Entry<K, V>> valueSet();
    public void clear();
    public boolean equals(Object o);
override final fun next() = nextShort() /* Returns the next value in the sequence without boxing. */
public abstract fun nextShort(): Short

// An iterator over a sequence of values of type `Int`. */
public abstract class IntIterator : Iterator<Int> {

    override fun next() = nextInt()

    /** Returns the next value in the sequence without boxing. */
    public abstract fun nextInt(): Int

    /** An iterator over a sequence of values of type `Long`. */
    public abstract class LongIterator : Iterator<Long> {

        override fun next() = nextLong()

        /** Returns the next value in the sequence without boxing. */
        public abstract fun nextLong(): Long

        // The last value in the progression.
        public val last: Long

        // The step of the progression.
        public val step: Long

        /** An iterator over a sequence of values of type `Double`. */
        public abstract class DoubleIterator : Iterator<Double> {

            override fun next() = nextDouble()

            /** Returns the next value in the sequence without boxing. */
            public abstract fun nextDouble(): Double

            // The last value in the progression.
            public val last: Double

            // The step of the progression.
            public val step: Double

            /** An iterator over a sequence of values of type `Float`. */
            public abstract class FloatIterator : Iterator<Float> {

                override fun next() = nextFloat()

                /** Returns the next value in the sequence without boxing. */
                public abstract fun nextFloat(): Float

                // The last value in the progression.
                public val last: Float

                // The step of the progression.
                public val step: Float

                /** An iterator over a sequence of values of type `Boolean`. */
                public abstract class BooleanIterator : Iterator<Boolean> {

                    override fun next() = nextBoolean()

                    /** Returns the next value in the sequence without boxing. */
                    public abstract fun nextBoolean(): Boolean

                    // The last value in the progression.
                    public val last: Boolean

                    // The step of the progression.
                    public val step: Boolean

                    /** An iterator over a sequence of values of type `Char`. */
                    public abstract class CharIterator : Iterator<Char> {

                        override fun next() = nextChar()

                        /** Returns the next value in the sequence without boxing. */
                        public abstract fun nextChar(): Char

                        // The last value in the progression.
                        public val last: Char

                        // The step of the progression.
                        public val step: Char

                        /** An iterator over a sequence of values of type `Int`. */
                        public abstract class IntProgression {

                            init {
                                if (step == 0) throw kotlin.IllegalArgumentException("Step must be non-zero.")
                                if (step == Int.MIN_VALUE) throw kotlin.IllegalArgumentException("Step must be greater than Int.MIN_VALUE to avoid overflow on negation.")
                            }

                            internal constructor
                                (start: Char, last: Char, val step: Int) : CharIterator() {

                                private var next = if (hasNext) start else last.toInt()
                                private var hasNext: Boolean = if (step > 0) start <= last else start >= last
                                override fun iterator(): CharIterator = CharProgressionIterator(first = start, last = last, step = step)
                                override fun hasNext(): Boolean = hasNext
                                override fun nextChar(): Char = getProgressionLastElement(start, step)
                            }

                            /**
                             * The first element in the progression.
                             */
                            public val first: Char = start

                            /**
                             * The step of the progression.
                             */
                            public val step: Int = step

                            /**
                             * The last element in the progression.
                             */
                            public val last: Char = last

                            /**
                             * The progression is empty.
                             */
                            public override fun isEmpty(): Boolean = if (step > 0) first <= last else first >= last

                            /**
                             * The progression.
                             */
                            public var progression: IntProgression = IntProgression(first = Int.MIN_VALUE, step = step)

                            /**
                             * Checks if the progression is empty.
                             */
                            public fun isEmpty(): Boolean = progression.isEmpty() || progression.iterator().hasNext()

                            override fun iterator(): CharIterator = this.iterator()

                            override fun hasNext(): Boolean = hasNext

                            override fun nextChar(): Char = getProgressionLastElement(hasNext = hasNext && hasNext)
override fun equals(other: Any?): Boolean =
    other is CharProgression && (isEmpty() && other.isEmpty()) ||
    first == other.first && last == other.last && step == other.step

override fun hashCode(): Int =
    if (isEmpty()) -1 else (31 * (31 * first.toInt() + last.toInt()) + step).toInt()
Represents a range of values (for example, numbers or characters).

See the [Kotlin language documentation](https://kotlinlang.org/docs/reference/ranges.html) for more information.

```kotlin
public interface ClosedRange<T: Comparable<T>> {
    /**
     * The minimum value in the range.
     */
    public val start: T

    /**
     * The maximum value in the range (inclusive).
     */
    public val endInclusive: T

    /**
     * Checks whether the specified [value] belongs to the range.
     */
    public operator fun contains(value: T): Boolean = value >= start && value <= endInclusive

    /**
     * Checks whether the range is empty.
     */
    public fun isEmpty(): Boolean = start > endInclusive
}
```

/*
 * Copyright 2010-2015 JetBrains s.r.o.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

package kotlin

/**
 * The type with only one value: the `Unit` object. This type corresponds to the `void` type in Java.
 */
public object Unit {
    override fun toString() = "kotlin.Unit"
}

/*
 * Copyright 2010-2015 JetBrains s.r.o.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

package kotlin.annotation

import kotlin.annotation.AnnotationTarget.*

/**
 * Contains the list of code elements which are the possible annotation targets
 */
public enum class AnnotationTarget {
    /**
     * Class, interface or object, annotation class is also included
     */
    CLASS,
    /** Annotation class only */
    ANNOTATION_CLASS,
    /** Generic type parameter (unsupported yet) */
    TYPE_PARAMETER,
    /** Property */
    PROPERTY,
    /** Field, including property's backing field */
    FIELD,
    /** Local variable */
    LOCAL_VARIABLE,
    /** Constructor only (primary or secondary) */
    CONSTRUCTOR,
    /** Type usage */
    TYPE,
    /** Any expression */
    EXPRESSION,
    /** File */
    FILE,
    /** Type alias */
    TYPEALIAS
}

/**
 * Contains the list of possible annotation retentions.
 * Determines how an annotation is stored in binary output.
 */
public enum class AnnotationRetention {
    /**
     * Annotation isn't stored in binary output
     */
    SOURCE,
    /** Annotation is stored in binary output, but invisible for reflection (default retention)
     */
    BINARY,
    /** Annotation is stored in binary output and visible for reflection
     */
    RUNTIME
}

/**
 * This meta-annotation indicates the kinds of code elements which are possible targets of an annotation.
 * If the target meta-annotation is not present on an annotation declaration, the annotation is applicable to the following elements:
 * 
 * @property allowedTargets list of allowed annotation targets
 */
@Target(AnnotationTarget.ANNOTATION_CLASS)
@MustBeDocumented
public annotation class Target(vararg val allowedTargets: AnnotationTarget)

/**
 * This meta-annotation determines that an annotation is a part of public API and therefore should be included in the
> shortArrayIterator(array)\n    \"CharArray\" -> charArrayIterator(array)\n    \"IntArray\" ->
    intArrayIterator(array)\n    \"LongArray\" -> longArrayIterator(array)\n    \"FloatArray\" ->
    floatArrayIterator(array)\n    \"DoubleArray\" -> doubleArrayIterator(array)\n    else -> throw

IllegalStateException("Unsupported type argument for arrayIterator:
\$type")
}

@JsName("booleanArrayIterator")
internal fun booleanArrayIterator(array: BooleanArray) =
object : BooleanIterator() {
    var index = 0
    override fun hasNext() = index < array.size
    override fun
    nextBoolean() = if (index < array.size) array[index++] else throw

NoSuchElementException("$index")
}

@JsName("byteArrayIterator")
internal fun byteArrayIterator(array: ByteArray) = object :
    ByteIterator() {
    var index = 0
    override fun hasNext() = index < array.size
    override fun
    nextByte() = if (index < array.size) array[index++] else throw

NoSuchElementException("$index")
}

@JsName("shortArrayIterator")
internal fun shortArrayIterator(array: ShortArray) = object :
    ShortIterator() {
    var index = 0
    override fun hasNext() = index < array.size
    override fun
    nextShort() = if (index < array.size) array[index++] else throw

NoSuchElementException("$index")
}

@JsName("charArrayIterator")
internal fun charArrayIterator(array: CharArray) = object :
    CharIterator() {
    var index = 0
    override fun hasNext() = index < array.size
    override fun
    nextChar() = if (index < array.size) array[index++] else throw

NoSuchElementException("$index")
}

@JsName("intArrayIterator")
internal fun intArrayIterator(array: IntArray) = object :
    IntIterator() {
    var index = 0
    override fun hasNext() = index < array.size
    override fun
    nextInt() = if (index < array.size) array[index++] else throw

NoSuchElementException("$index")
}

@JsName("floatArrayIterator")
internal fun floatArrayIterator(array: FloatArray) = object :
    FloatIterator() {
    var index = 0
    override fun hasNext() = index < array.size
    override fun
    nextFloat() = if (index < array.size) array[index++] else throw

NoSuchElementException("$index")
}

@JsName("doubleArrayIterator")
internal fun doubleArrayIterator(array: DoubleArray) = object :
    DoubleIterator() {
    var index = 0
    override fun hasNext() = index < array.size
    override fun
    nextDouble() = if (index < array.size) array[index++] else throw

NoSuchElementException("$index")
}

@JsName("longArrayIterator")
internal fun longArrayIterator(array: LongArray) = object :
    LongIterator() {
    var index = 0
    override fun hasNext() = index < array.size
    override fun
    nextLong() = if (index < array.size) array[index++] else throw

NoSuchElementException("$index")
}

@JsName("PropertyMetadata")
internal class PropertyMetadata(@JsName("callableName") val name: String)

@JsName("noWhenBranchMatched")
internal fun noWhenBranchMatched(): Nothing = throw

@JsName("BoxedChar")
internal class BoxedChar(val c: Int) : Comparable<Int> {
    override
    fun equals(other: Any?): Boolean {
        return other is BoxedChar && c == other.c
    }

    override fun
    hashCode(): Int {
        return c
    }

    override fun
    toString(): String {
        return js("this.c").unsafeCast<Char>().toString()
    }

    override fun
    compareTo(other: Int): Int {
        return c
    }

    @JsName("valueOf")
    public fun valueOf(): Int {
        return c
    }
}

@kotlin.internal.InlineOnly
internal inline fun <T> concat(args: Array<T>): T
srcLen && index < dstLen) dst[index] = src[index++]
    return dst
}

internal fun arrayCopyResize(source: dynamic, newSize: Int, defaultValue: Any?): dynamic {
    val result = source.slice(0, newSize)
    copyArrayType(source, result)
    var index = source.length
    if (newSize > index) {
        result.length = newSize
        while (index < newSize) result[index++] = defaultValue
    }
    return result
}

internal fun <T> arrayPlusCollection(array: dynamic, collection: Collection<T>): dynamic {
    val result = array.slice()
    result.length += collection.size
    copyArrayType(array, result)
    var index = array.length
    for (element in collection) result[index++] = element
    return result
}

internal fun <T> fillFromCollection(dst: dynamic, startIndex: Int, collection: Collection<T>): dynamic {
    var index = startIndex
    for (element in collection) dst[index++] = element
    return dst
}

internal inline fun copyArrayType(from: dynamic, to: dynamic) {
    if (from.`$type$` !== undefined) {
        to.`$type$` = from.`$type$`
    }
}

internal inline fun jsIsType(obj: dynamic, jsClass: dynamic) = js("Kotlin").isType(obj, jsClass)

package kotlin

/**
 * Creates a Char with the specified code.
 *
 * @sample samples.text.Chars.charFromCode
 */
@OptIn(ExperimentalUnsignedTypes::class)
@ExperimentalStdlibApi
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public actual inline fun Char(code: UShort): Char {
    return code.toInt().toChar()
}

package kotlin.coroutines

import kotlin.coroutines.intrinsics.COROUTINE_SUSPENDED

@SinceKotlin("1.3")
@JsName("CoroutineImpl")
internal abstract class CoroutineImpl(private val resultContinuation: Continuation<Any?>) : Continuation<Any?> {
    protected var state = 0
    protected var exceptionState = 0
    protected var result: Any? = null
    protected var exception: Throwable? = null
    protected var finallyPath: Array<Int>? = null

    public override val context: CoroutineContext = resultContinuation.context

    private var intercepted_: Continuation<Any?>? = null

    public fun intercepted(): Continuation<Any?> =
            intercepted_ ?: (context[ContinuationInterceptor]!!.interceptContinuation(this) ?: this)
                .also { intercepted_ = it }

    override fun resumeWith(result: Result<Any?>) {
        var current = this
        var currentResult: Any? = result.getOrNull()
        var currentException: Throwable? = result.exceptionOrNull()

        // This loop unrolls recursion in current.resumeWith(param) to make saner and shorter stack traces on resume
        while (true) {
            with(current) {
                val completion = resultContinuation

                // Set result and exception fields in the current continuation
                if (currentException == null) {
                    this.result = currentResult
                } else {
                    state = exceptionState
                    exception = currentException
                }

                try {
                    val outcome = doResume()
                    if (outcome === COROUTINE_SUSPENDED)
                        return
                    currentResult = outcome
                    currentException = null
                } catch (exception: dynamic) { // Catch all exceptions
                    currentResult = null
                    currentException = exception.unsafeCast<Throwable>()
                }

                releaseIntercepted() // this state machine instance is terminating

                if (completion is CoroutineImpl) {
                    // unrolling recursion via loop
                    current = completion
                } else {
                    // top-level completion reached -- invoke and return
                    currentException?.let {
                        completion.resumeWithException(it)
                    } ?: completion.resume(resultContinuation)
                }
            }
        }
    }

    private fun releaseIntercepted() {
        val intercepted = intercepted_
        if (intercepted != null && intercepted !== this) {
            context[ContinuationInterceptor]!!.releaseInterceptedContinuation(intercepted)
            this.intercepted_ = CompletedContinuation // just in case
        }
        else { // protected fun releaseIntercepted() { // catch (exception: dynamic) {
            // Catch all exceptions
            currentResult = null
            currentException = exception.unsafeCast<Throwable>()
        }

        releaseIntercepted() { // this state machine instance is terminating
            if (completion is CoroutineImpl) {
                // unrolling recursion via loop
                current = completion
            } else {
                completion.resumeWithException(it)

            }
        }
    }
}

internal object CompletedContinuation : Continuation<Any?> {
    override val context: CoroutineContext
        get() = error("This continuation is already complete")
    override fun resumeWith(result: Result<Any?>) {
        error("This continuation is already complete")
    }
    override fun toString(): String = "\047This continuation is already complete\047"
}

@author Copyright 2010-2018 JetBrains s.r.o. and
Kotlin Programming Language contributors.* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

*package kotlin

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin

/**
 * Creates a Char with the specified code.
 */
@OptIn(ExperimentalUnsignedTypes::class)
@ExperimentalStdlibApi
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public actual inline fun Char(code: UShort): Char {
    return code.toInt().toChar()
}

package kotlin.coroutines

import kotlin.coroutines.intrinsics.COROUTINE_SUSPENDED

@SinceKotlin("1.3")
@JsName("CoroutineImpl")
internal abstract class CoroutineImpl(private val resultContinuation: Continuation<Any?>) : Continuation<Any?> {
    protected var state = 0
    protected var exceptionState = 0
    protected var result: Any? = null
    protected var exception: Throwable? = null
    protected var finallyPath: Array<Int>? = null

    public override val context: CoroutineContext = resultContinuation.context

    private var intercepted_: Continuation<Any?>? = null

    public fun intercepted(): Continuation<Any?> =
            intercepted_ ?: (context[ContinuationInterceptor]!!.interceptContinuation(this) ?: this)
                .also { intercepted_ = it }

    override fun resumeWith(result: Result<Any?>) {
        var current = this
        var currentResult: Any? = result.getOrNull()
        var currentException: Throwable? = result.exceptionOrNull()

        // This loop unrolls recursion in current.resumeWith(param) to make saner and shorter stack traces on resume
        while (true) {
            with(current) {
                val completion = resultContinuation

                // Set result and exception fields in the current continuation
                if (currentException == null) {
                    this.result = currentResult
                } else {
                    state = exceptionState
                    exception = currentException
                }

                try {
                    val outcome = doResume()
                    if (outcome === COROUTINE_SUSPENDED)
                        return
                    currentResult = outcome
                    currentException = null
                } catch (exception: dynamic) { // Catch all exceptions
                    currentResult = null
                    currentException = exception.unsafeCast<Throwable>()
                }

                releaseIntercepted() // this state machine instance is terminating

                if (completion is CoroutineImpl) {
                    // unrolling recursion via loop
                    current = completion
                } else {
                    // top-level completion reached -- invoke and return
                    currentException?.let {
                        completion.resumeWithException(it)
                    } ?: completion.resume(resultContinuation)
                    return
                }
            }
        }
    }

    private fun releaseIntercepted() {
        val intercepted = intercepted_
        if (intercepted != null && intercepted !== this) {
            context[ContinuationInterceptor]!!.releaseInterceptedContinuation(intercepted)
            this.intercepted_ = CompletedContinuation // just in case
        }
        else { // protected fun releaseIntercepted() { // catch (exception: dynamic) {
            // Catch all exceptions
            currentResult = null
            currentException = exception.unsafeCast<Throwable>()
        }

        releaseIntercepted() { // this state machine instance is terminating
            if (completion is CoroutineImpl) {
                // unrolling recursion via loop
                current = completion
            } else {
                completion.resumeWithException(it)

            }
        }
    }
}

internal object CompletedContinuation : Continuation<Any?> {
    override val context: CoroutineContext
        get() = error("This continuation is already complete")
    override fun resumeWith(result: Result<Any?>) {
        error("This continuation is already complete")
    }
    override fun toString(): String = "\047This continuation is already complete\047"
}

@author Copyright 2010-2018 JetBrains s.r.o. and
Kotlin Programming Language contributors.* Use of this source code is governed by the Apache 2.0 license that can be found in the
public inline class Result<out T> @PublishedApi internal constructor(
    @PublishedApi
    internal val value:
    Any?
) : Serializable {
    // discovery

    /**
     * Returns `true` if this instance represents a successful
     * outcome.
     */
    public val isSuccess: Boolean get() = value !is Failure

    /**
     * Returns `true` if this instance represents a failed outcome.
     */
    public val isFailure: Boolean get() = value is Failure

    // value & exception
    // retrieval

    /**
     * Returns the encapsulated value if this instance represents [success][Result.isSuccess] or
     * `null` if it is [failure][Result.isFailure].
     */
    @InlineOnly
    public inline fun getOrNull(): T? =
        when {
            isFailure -> null
            else -> value as T
        }

    /**
     * Returns the encapsulated [Throwable] exception if this instance represents [failure][isFailure] or `null` if it is [success][isSuccess].
     */
    public fun exceptionOrNull(): Throwable? =
        when (value) {
            is Failure ->
                value.exception
            else -> null
        }

    /**
     * Returns a string `Success(v)` if this instance represents [success][Result.isSuccess] where `v` is a string representation of the value or a string `Failure(x)` if it is [failure][isFailure] where `x` is a string representation of the exception.
     */
    public override fun toString(): String =
        when (value) {
            is Failure -> value.toString() // "Failure($exception)"
            else -> "Success($value)"
        }

    // companion with constructors

    /**
     * Companion object for [Result] class that contains its constructor functions
     * [success] and [failure].
     */
    public companion object {
        /**
         * Returns an instance that encapsulates the given [value] as successful value.
         */
        @Suppress("INAPPLICABLE_JVM_NAME")
        @InlineOnly
        @JvmName("success")
        public inline fun <T> success(value: T): Result<T> =
            Result(value)

        /**
         * Returns an instance that encapsulates the given [exception] as failure.
         */
        @Suppress("INAPPLICABLE_JVM_NAME")
        @InlineOnly
        @JvmName("failure")
        public inline fun <T> failure(exception: Throwable): Result<T> =
            Result.createFailure(exception)
    }

    internal class Failure(
        @JvmField
        val exception: Throwable
    ) : Serializable {
        override fun equals(other: Any?): Boolean = other is Failure && exception == other.exception
        override fun hashCode(): Int = exception.hashCode()
        override fun toString(): String = "Failure($exception)"
    }

    // --
    // extensions ---

    /**
     * Returns the encapsulated value if this instance represents [success][Result.isSuccess] or
exceptionOrNull()) {\n    null -> this\n    else -> runCatching { transform(exception) }\n}

/**
 * Performs the given [action] on the encapsulated [Throwable] exception if
 * this instance represents [failure][Result.isFailure].\n * Returns the original `Result` unchanged.
 */
@InlineOnly\n@SinceKotlin("1.3")\npublic inline fun <T> Result<T>.onFailure(action: (exception: Throwable) -> Unit): Result<T> {\n    contract { \n        callsInPlace(action, InvocationKind.AT_MOST_ONCE)\n    }\n    exceptionOrNull()?.let {\n        action(it)\n    }\n    return this\n}

/**
 * Performs the given [action] on the
 * encapsulated value if this instance represents [success][Result.isSuccess].\n * Returns the original `Result` unchanged.
 */
@InlineOnly\n@SinceKotlin("1.3")\npublic inline fun <T> Result<T>.onSuccess(action: (value: T) -> Unit): Result<T> {\n    contract { \n        callsInPlace(action, InvocationKind.AT_MOST_ONCE)\n    }\n    if (isSuccess) action(value as T)\n    return this\n}

// -------------------

**Coroutine Context**

CoroutineContext get() { throw NotImplementedError("Implemented as intrinsic")

Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

@SinceKotlin("1.3")
@InlineOnly
public actual inline fun <R, P, T> (suspend R.(P) -> T).startCoroutineUninterceptedOrReturn(
    receiver: R,
    param: P,
    completion: Continuation<T>
): Any? = this.asDynamic()(receiver, param, completion, false)

@SinceKotlin("1.3")
@InlineOnly
public actual inline fun <T> (suspend () -> T).startCoroutineUninterceptedOrReturn(
    completion: Continuation<T>
): Any? = this.asDynamic()(completion, false)

*COROUTINE_SUSPENDED* * Starts a coroutine without a receiver and with result type [T].*

*This function creates and starts a new, fresh instance of suspsendable computation every time it is invoked.*

*The completion continuation is invoked when the coroutine completes with a result or an exception.*

*This function creates and starts a new, fresh instance of suspsendable computation every time it is invoked.*

*The completion continuation is invoked when the coroutine completes with a result or an exception.*

*Obtains the current continuation instance inside suspend functions and suspends the currently running coroutine.*

*In this function both [Completion.resume] and [Completion.resumeWithException] can be used either synchronously or asynchronously in the same stack-frame where the suspension function is run or asynchronously later in the same thread or on a different thread of execution. Subsequent invocation of any resume function will produce an [IllegalStateException].*

*This function is designed to be used from inside of [suspendCoroutineUninterceptedOrReturn] to resume the execution of the suspended coroutine without a receiver and with result type [T] and executes it until its first suspension.*

*Returns the result of the coroutine or throws its exception if it does not suspend or [COROUTINE_SUSPENDED] if it suspends.*

*In the latter case, the completion continuation is invoked when the coroutine completes with a result or an exception.*

*The coroutine is started directly in the invoker's thread without going through the [ContinuationInterceptor] that might be present in the completion's [CoroutineContext]. It is the invoker's responsibility to ensure that a proper invocation context is established.*

*This function is designed to be used from inside of [suspendCoroutineUninterceptedOrReturn] to resume the execution of the suspended coroutine using a reference to the suspending function.*

*This function is designed to be used from inside of [suspendCoroutineUninterceptedOrReturn] to resume the execution of the suspended coroutine using a reference to the suspending function.*

*This function is designed to be used from inside of [suspendCoroutineUninterceptedOrReturn] to resume the execution of the suspended coroutine using a reference to the suspending function.*

*This function is designed to be used from inside of [suspendCoroutineUninterceptedOrReturn] to resume the execution of the suspended coroutine using a reference to the suspending function.*

*This function is designed to be used from inside of [suspendCoroutineUninterceptedOrReturn] to resume the execution of the suspended coroutine using a reference to the suspending function.*
receiver and with result type [T]).
* This function creates a new, fresh instance of suspendable computation every
time it is invoked.
* To start executing the created coroutine, invoke `resume(Unit)` on the returned
[Continuation] instance.
* The [completion] continuation is invoked when coroutine completes with result or
exception.
* This function returns unintercepted continuation.
* Invocation of `resume(Unit)` starts coroutine immediately in the invoker's call stack without going through the
* [ContinuationInterceptor] that might be present in the completion's [CoroutineContext].
* It is the invoker's responsibility to ensure that a proper invocation context is established.
* Note that [completion] of this function may get invoked in an arbitrary context.
* [Continuation.intercepted] can be used to acquire the intercepted continuation.
* Invocation of `resume(Unit)` on intercepted continuation guarantees that execution of
* both the coroutine and [completion] happens in the invocation context established by
* [ContinuationInterceptor].
* Repeated invocation of any resume function on the resulting continuation corrupts
* execution of state machine of the coroutine and may result in arbitrary behaviour or
* exception.
* This function creates a new, fresh instance of suspendable computation every time it is invoked.
* To start executing the created coroutine, invoke `resume(Unit)` on the returned
[Continuation] instance.
* The [completion] continuation is invoked when coroutine completes with result or
exception.
* This function returns unintercepted continuation.
* Invocation of `resume(Unit)` starts coroutine immediately in the invoker's call stack without going through the
* [ContinuationInterceptor] that might be present in the completion's [CoroutineContext].
* It is the invoker's responsibility to ensure that a proper invocation context is established.
* Note that [completion] of this function may get invoked in an arbitrary context.
* [Continuation.intercepted] can be used to acquire the intercepted continuation.
* Invocation of `resume(Unit)` on intercepted continuation guarantees that execution of
* both the coroutine and [completion] happens in the invocation context established by
* [ContinuationInterceptor].
* Repeated invocation of any resume function on the resulting continuation corrupts
* execution of state machine of the coroutine and may result in arbitrary behaviour or exception.
* This function creates a new, fresh instance of suspendable computation every time it is invoked.
* To start executing the created coroutine, invoke `resume(Unit)` on the returned
[Continuation] instance.
* The [completion] continuation is invoked when coroutine completes with result or
exception.
* This function returns unintercepted continuation.
* Invocation of `resume(Unit)` starts coroutine immediately in the invoker's call stack without going through the
* [ContinuationInterceptor] that might be present in the completion's [CoroutineContext].
* It is the invoker's responsibility to ensure that a proper invocation context is established.
* Note that [completion] of this function may get invoked in an arbitrary context.
* [Continuation.intercepted] can be used to acquire the intercepted continuation.
* Invocation of `resume(Unit)` on intercepted continuation guarantees that execution of
* both the coroutine and [completion] happens in the invocation context established by
* [ContinuationInterceptor].
* Repeated invocation of any resume function on the resulting continuation corrupts
* execution of state machine of the coroutine and may result in arbitrary behaviour or exception.
* This function creates a new, fresh instance of suspendable computation every time it is invoked.
* To start executing the created coroutine, invoke `resume(Unit)` on the returned
[Continuation] instance.
* The [completion] continuation is invoked when coroutine completes with result or
exception.
* This function returns unintercepted continuation.
* Invocation of `resume(Unit)` starts coroutine immediately in the invoker's call stack without going through the
* [ContinuationInterceptor] that might be present in the completion's [CoroutineContext].
* It is the invoker's responsibility to ensure that a proper invocation context is established.
* Note that [completion] of this function may get invoked in an arbitrary context.
* [Continuation.intercepted] can be used to acquire the intercepted continuation.
* Invocation of `resume(Unit)` on intercepted continuation guarantees that execution of
* both the coroutine and [completion] happens in the invocation context established by
* [ContinuationInterceptor].
* Repeated invocation of any resume function on the resulting continuation corrupts
* execution of state machine of the coroutine and may result in arbitrary behaviour or exception.

---

```kotlin
@SinceKotlin("1.3")
public actual fun <T> (suspend () -> T).createCoroutineUnintercepted(
    completion: Continuation<T>
): Continuation<Unit> =
    // Kotlin/JS suspend lambdas have an extra parameter
    // When `suspended` is true the continuation is created, but not executed
    this.asDynamic()() ?: { this.asDynamic()() suspend R().() -> T).createCoroutineUnintercepted(
        receiver: R,
        completion: Continuation<T>
    ): Continuation<Unit> =
        // Kotlin/JS suspend lambdas have an extra parameter
        // When `suspended` is true the continuation is created, but not executed
        this.asDynamic()() ?: { this.asDynamic()() ?
```

---

```kotlin
private inline fun <T> createCoroutineFromSuspendFunction(
    completion: Continuation<T>,
    crossinline block: () -> Any?
): Continuation<Unit> {
    return object
        : CoroutineImpl(completion as Continuation<Any?>) {
            override fun doResume(): Any? { exception?.let { throw it } return block() }`
```
definedExternally

@Suppress("NOTHING_TO_INLINE")
internal inline fun isArrayish(o: dynamic) = js("Kotlin").isArrayish(o)

package kotlin

// TODO: remove primary constructors, make all secondary KT-22055

@Suppress("USELESS_ELVIS_RIGHT_IS_NULL")
public

public actual open class Error actual constructor(message: String?, cause: Throwable?) : Throwable(message, cause ?: null)

public actual open class Exception actual constructor(message: String?, cause: Throwable?) : Throwable(message, cause ?: null)

public actual open class RuntimeException actual constructor(message: String?) : Exception(message)

public actual open class IllegalArgumentException actual constructor(message: String?, cause: Throwable?) : RuntimeException(message, cause)

public actual open class IllegalStateException actual constructor(message: String?, cause: Throwable?) : RuntimeException(message, cause)

public actual open class IndexOutOfBoundsException actual constructor(message: String?) : RuntimeException(message)

public actual open class ConcurrentModificationException actual constructor(message: String?, cause: Throwable?) : RuntimeException(message, cause)

public actual open class UnsupportedOperationException actual constructor(message: String?, cause: Throwable?) : RuntimeException(message, cause)

public actual open class NumberFormatException actual constructor(message: String?) : IllegalArgumentException(message)

public actual open class NullPointerException actual constructor(message: String?) : RuntimeException(message)

public actual open class ClassCastException actual constructor(message: String?) : RuntimeException(message)

public actual open class AssertionError

public actual open class NoSuchElementException actual constructor(message: String?) : RuntimeException(message)

public actual open class ArithmeticException actual constructor(message: String?) : RuntimeException(message)

public actual open class NoWhenBranchMatchedException actual constructor(message: String?, cause: Throwable?) : RuntimeException(message, cause)

public actual open class UninitializedPropertyAccessException actual constructor(message: String?, cause: Throwable?) : RuntimeException(message, cause)

ClassCastException actual constructor(message: String?) : RuntimeException(message)

NoSuchElementException actual constructor(message: String?) : RuntimeException(message)

NullPointerException actual constructor(message: String?) : RuntimeException(message)

NullPointerException actual constructor(message: String?) : RuntimeException(message)

NullPointerException actual constructor(message: String?) : RuntimeException(message)

NullPointerException actual constructor(message: String?) : RuntimeException(message)

NullPointerException actual constructor(message: String?) : RuntimeException(message)

NullPointerException actual constructor(message: String?) : RuntimeException(message)

NullPointerException actual constructor(message: String?) : RuntimeException(message)

NullPointerException actual constructor(message: String?) : RuntimeException(message)

NullPointerException actual constructor(message: String?) : RuntimeException(message)
governed by the Apache 2.0 license * that can be found in the license/LICENSE.txt file.

*\n
*/

package kotlin

@kotlin.internal.InlineOnly
internal inline fun jsDeleteProperty(obj: Any, property: Any) {
    js("delete obj[property]")
}

@kotlin.internal.InlineOnly
internal inline fun jsBitwiseOr(lhs: Any?, rhs: Any?): Int =
    js("lhs | rhs").unsafeCast<Int>().unsafeCast<Int>()

@file:Suppress("UNUSED_PARAMETER")

package kotlin

/**
 * Returns a bit representation of the specified floating-point value as [Long]\n * according to the IEEE 754 floating-point \"double format\" bit layout.
 */
public actual fun Double.toBits(): Long = definedExternally

/**
 * Returns a bit representation of the specified floating-point value as [Int]\n * according to the IEEE 754 floating-point \"single format\" bit layout.
 */
public actual fun Float.toBits(): Int = definedExternally

*/

import kotlin.reflect.KClass

@PublishedApi
internal fun <T : Annotation> KClass<*>.findAssociatedObject(@Suppress("UNUSED_PARAMETER") annotationClass: KClass<T>): Any? {
    return null
}

package kotlin.text

/**
 * Returns a string representation of this [Long] value in the specified [radix].
 * @throws IllegalArgumentException when [radix] is not a valid radix for number to string conversion.
 */
public actual fun Long.toString(radix: Int): String = asDynamic().toString(checkRadix(radix))

*/

Open Source Used In security_automation identity-guest-cs 1582
Language contributors. * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file. *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *\n *} 

Open Source Used In security_automation identity-guest-cs 1583
override fun isEmpty(): Boolean = this@asList.isEmpty()
    override fun contains(element: ULong): Boolean = this@asList.contains(element)
    override fun get(index: Int): ULong {
        AbstractList.checkElementIndex(index, size)
        return this@asList[index]
    }
    override fun lastIndexOf(element: ULong): Int {
        @Suppress("USELESS_CAST")
        if ((element as Any?) !is ULong) return -1
        return this@asList.lastIndexOf(element)
    }
}

/**
 * Returns a [List] that wraps the original array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public actual fun UByteArray.asList(): List<UByte> {
    return object : AbstractList<UByte>(), RandomAccess {
        override val size: Int get() = this@asList.size
        override fun isEmpty(): Boolean = this@asList.isEmpty()
        override fun contains(element: UByte): Boolean = this@asList.contains(element)
        override fun get(index: Int): UByte {
            AbstractList.checkElementIndex(index, size)
            return this@asList[index]
        }
        override fun indexOf(element: UByte): Int {
            @Suppress("USELESS_CAST")
            if ((element as Any?) !is UByte) return -1
            return this@asList.indexOf(element)
        }
        override fun lastIndexOf(element: UByte): Int {
            @Suppress("USELESS_CAST")
            if ((element as Any?) !is UByte) return -1
            return this@asList.lastIndexOf(element)
        }
    }
}

/**
 * Returns a [List] that wraps the original array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public actual fun UShortArray.asList(): List<UShort> {
    return object : AbstractList<UShort>(), RandomAccess {
        override val size: Int get() = this@asList.size
        override fun isEmpty(): Boolean = this@asList.isEmpty()
        override fun contains(element: UShort): Boolean = this@asList.contains(element)
        override fun get(index: Int): UShort {
            AbstractList.checkElementIndex(index, size)
            return this@asList[index]
        }
        override fun indexOf(element: UShort): Int {
            @Suppress("USELESS_CAST")
            if ((element as Any?) !is UShort) return -1
            return this@asList.indexOf(element)
        }
        override fun lastIndexOf(element: UShort): Int {
            @Suppress("USELESS_CAST")
            if ((element as Any?) !is UShort) return -1
            return this@asList.lastIndexOf(element)
        }
    }
}
JQuery = asDynamic().draggable(params)

@Deprecated("Use declarations from 'https://bintray.com/kotlin/js-externals/kotlin-js-jquery' package instead.", level = DeprecationLevel.ERROR)

public inline fun JQuery.selectable(): JQuery = asDynamic().selectable()

/*
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin

public actual fun interface Comparator<T> {
    @JsName("compare")
    public actual fun compare(a: T, b: T): Int
}

/*
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.js

import kotlin.annotation.AnnotationTarget.*

@Target(FUNCTION)
@Deprecated("Use inline extension function with body using dynamic")
public annotation class nativeGetter

@Target(FUNCTION)
@Deprecated("Use inline extension function with body using dynamic")
public annotation class nativeSetter

@Target(FUNCTION)
@Deprecated("Use inline extension function with body using dynamic")
public annotation class nativeInvoke

@Target(CLASS, FUNCTION, PROPERTY)
internal annotation class library(public val name: String = "")

@Target(CLASS)
internal annotation class marker

/**
 * Gives a declaration (a function, a property or a class) specific name in JavaScript.
 * This may be useful in the following cases:
 * * There are two functions for which the compiler gives same name in JavaScript, you can mark one with `@JsName(...)` to prevent the compiler from reporting error.
 * * You are writing a JavaScript library in Kotlin. The compiler produces mangled names for functions with parameters, which is unnatural for usual JavaScript developer.
 * * For some reason you want to rename declaration, e.g. there's common term in JavaScript for a concept provided by the declaration, which in uncommon in Kotlin.
 * For example:
 * ```kotlin
 * class Person(val name: String) {
 *     fun hello() {
 *         println("Hello $name!")
 *     }
 *
 *     @JsName("helloWithGreeting")
 *     fun hello(greeting: String) {
 *         println("$greeting $name!")
 *     }
 * }
 * ```
 * @property name the name which compiler uses both for declaration itself and for all references to the declaration.
 * It's required to denote a valid JavaScript identifier.
 * You can put `@JsName(...)` on functions you want to be available from JavaScript.
 * For some reason you want to rename declaration, e.g. there's common term in JavaScript for a concept provided by the declaration, which in uncommon in Kotlin.
 * For example:
 * ```kotlin
 * class Person(val name: String) {
 *     fun hello() {
 *         println("Hello $name!")
 *     }
 *
 *     @JsName("helloWithGreeting")
 *     fun hello(greeting: String) {
 *         println("$greeting $name!")
 *     }
 * }
 * ```
 */
@Retention(AnnotationRetention.BINARY)
@Target(CLASS, FUNCTION, PROPERTY, CONSTRUCTOR, PROPERTY_GETTER, PROPERTY_SETTER)
public annotation class JsName(actual val name: String)

/**
 * Denotes an `external` declaration that must be imported from native JavaScript library.
 * The compiler produces the code relevant for the target module system, for example, in case of CommonJS, it will import the declaration via the `require(...)` function.
 * The annotation can be used on top-level external declarations (classes, properties, functions) and files.
 * In case of file (which can't be `external`) the following rule applies: all the declarations in a file must be `external`. By applying `@JsModule(...)` on a file you tell the compiler to import a JavaScript object that contain all the declarations from the file.
 * Example:
 * ```kotlin
 * class Person(val name: String) {
 *     fun hello() {
 *         println("Hello $name!")
 *     }
 *
 *     @JsModule("jquery")
 *     fun helloWithGreeting(greeting: String) {
 *         println("$greeting $name!")
 *     }
 * }
 * ```
 * It's not interpreted by the Kotlin compiler, it's passed as is to the target module system.
 * @see JsNonModule
 */
@Retention(AnnotationRetention.BINARY)
@Target(CLASS, PROPERTY, FUNCTION, FILE)
public annotation class JsModule(val import: String)

/**
 * Adds prefix to `external`
declarations in a source file.

JavaScript does not have a concept of packages (namespaces). They are usually emulated by nested objects. The compiler turns references to `external` declarations either to plain unprefixed names (in case of plain modules) or to plain imports. However, if a JavaScript library provides its declarations in packages, you won't be satisfied with this.

You can tell the compiler to generate additional prefix before references to `external` declarations using the `@JsQualifier(...)` annotation.

Note that a file marked with the `@JsQualifier(...)` annotation can't contain non-`external` declarations.

Example of valid qualifiers are: `foo`, `bar.Baz`, `_.$0.123`.

```javascript
@file:JsQualifier("my.jsPackageName")
package some.kotlinPackage

external fun foo(x: Int)

external fun bar(): String
```

@property value the qualifier to add to the declarations in the generated code. It must be a sequence of valid JavaScript identifiers separated by the `.` character.

Examples of valid qualifiers are: `foo`, `bar.Baz`, `_.$0.123`.

Annotations:
- `@Retention(AnnotationRetention.BINARY)`
- `@Target(AnnotationTarget.FILE)`
- `@JsQualifier(val: String)`

```kotlin
@file:JsQualifier("my.jsPackageName")
package some.kotlinPackage

external fun foo(x: Int)

external fun bar(): String
```

### Annotation `@JsQualifier(val: String)`

#### Attributes:
- `value`: String

#### Usage:
- Exports the `value` attribute.
- The `value` attribute must be a sequence of valid JavaScript identifiers separated by the `.` character.
- Examples of valid qualifiers are: `foo`, `bar.Baz`, `_.$0.123`.

#### Example:

```kotlin
@file:JsQualifier("my.jsPackageName")
package some.kotlinPackage

external fun foo(x: Int)

external fun bar(): String
```

---

@Retention(AnnotationRetention.BINARY)
@Target(AnnotationTarget.FILE)
public annotation class JsQualifier(val: String)

```
@file:JsQualifier("my.jsPackageName")
package some.kotlinPackage

external fun foo(x: Int)

external fun bar(): String
```

### Annotation `@JsQualifier(val: String)`

#### Attributes:
- `value`: String

#### Usage:
- Exports the `value` attribute.
- The `value` attribute must be a sequence of valid JavaScript identifiers separated by the `.` character.
- Examples of valid qualifiers are: `foo`, `bar.Baz`, `_.$0.123`.

#### Example:

```kotlin
@file:JsQualifier("my.jsPackageName")
package some.kotlinPackage

external fun foo(x: Int)

external fun bar(): String
```

---

@Retention(AnnotationRetention.BINARY)
@Target(AnnotationTarget.FILE)
public annotation class JsQualifier(val: String)

```
@file:JsQualifier("my.jsPackageName")
package some.kotlinPackage

external fun foo(x: Int)

external fun bar(): String
```

### Annotation `@JsQualifier(val: String)`

#### Attributes:
- `value`: String

#### Usage:
- Exports the `value` attribute.
- The `value` attribute must be a sequence of valid JavaScript identifiers separated by the `.` character.
- Examples of valid qualifiers are: `foo`, `bar.Baz`, `_.$0.123`.

#### Example:

```kotlin
@file:JsQualifier("my.jsPackageName")
package some.kotlinPackage

external fun foo(x: Int)

external fun bar(): String
```
private class SubList<E>(private val list: AbstractMutableList<E>, private val fromIndex: Int, toIndex: Int) : 
AbstractMutableList<E>(), RandomAccess {
    override fun add(index: Int, element: E) {
        AbstractList.checkPositionIndex(index, _size)
        list.add(fromIndex + index, element)
        override fun fun get(index: Int): E {
            AbstractList.checkElementIndex(index, _size)
            return list[fromIndex + index]
        }
        override fun removeAt(index: Int): E {
            AbstractList.checkElementIndex(index, _size)
            val result = list.removeAt(fromIndex + index)
            _size--
            return result
        }
        override fun set(index: Int, element: E): E {
            AbstractList.checkElementIndex(index, _size)
            return list.set(fromIndex + index, element)
        }
        override val size: Int get() = _size
        internal override fun checkIsMutable(): Unit = list.checkIsMutable()
    }
}

/*
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the
 * license/LICENSE.txt file.
 */

/*
 * Based on GWT AbstractMap
 * Copyright 2007 Google Inc.
 */

package kotlin.collections

/**
 * Provides a skeletal implementation of the [MutableMap] interface.
 * The implementor is required to implement [entries] property, which should return mutable set of map entries, and
 * [put] function.
 * @param K the type of map keys. The map is invariant in its key type.
 * @param V the type of map values. The map is invariant in its value type.
 */
public actual abstract class AbstractMutableMap<K, V> 
checked constructor() : AbstractMap<K, V>(), MutableMap<K, V> {
    /**
     */
    internal open class SimpleEntry<K, V>(override
        val key: K, value: V) : MutableMap.MutableEntry<K, V> {
        constructor(entry: Map.Entry<K, V>) :
            this(entry.key, entry.value)
        private var _value = value
        override val value: V get() = _value
        override fun setValue(newValue: V): V {
            // Should check if the map containing this entry is mutable.
            // However, to not increase entry memory footprint it might be worthwhile not to check it here and //
            // force subclasses that implement `build()` (freezing) operation to implement their own `MutableEntry`.
            this@AbstractMutableMap.checkIsMutable()
            val oldValue = this._value
            this._value = newValue
            return oldValue
        }
        override fun hashCode(): Int = entryHashCode(this)
        override fun toString(): String = entryToString(this)
        override fun equals(other: Any?): Boolean = entryEquals(this,
            other)
    }

    // intermediate abstract class to workaround KT-43321
    internal abstract class 
        AbstractEntrySet<E : Map.Entry<K, V>, K, V> : AbstractMutableSet<E>() {
        final override fun contains(element: E): Boolean = throw
            UnsupportedOperationException("Add is not supported on keys")
        }

    actual override fun put(key: K, value: V): V?
    actual override fun putAll(from: Map<out K, V>) {
        checkIsMutable()
        for ((key, value) in from) {
            put(key, value)
        }
    }

    actual override fun get(index: Int): V {
        AbstractList.checkElementIndex(index, _size)
        return result
    }

    actual override fun removeAt(index: Int): E {
        AbstractList.checkElementIndex(index, _size)
        val result = list.removeAt(fromIndex + index)
        _size--
        return result
    }

    actual override fun set(index: Int, element: E): E {
        AbstractList.checkElementIndex(index, _size)
        return list.set(fromIndex + index, element)
    }

    actual override val size: Int get() = _size
    internal override fun checkIsMutable(): Unit = list.checkIsMutable()
}

Open Source Used In security_automation identity-guest cs 1588
add(element: V): Boolean = throw UnsupportedOperationException("Add is not supported on values")

override fun clear() = this@AbstractMutableMap.clear()

override fun contains(element: V): Boolean = containsValue(element)

override operator fun iterator(): MutableIterator<V> {
    val entryIterator = entries.iterator()
    return object : MutableIterator<V> {
        override fun hasNext(): Boolean = entryIterator.hasNext()
        override fun next(): V = entryIterator.next().value
        override fun remove() = entryIterator.remove()
    }
}

override val size: Int get() = this@AbstractMutableMap.size

// TODO: should we implement them this way? Currently it's unspecified in JVM
override fun equals(other: Any?): Boolean {
    if (this === other) return true
    if (other !is Collection<*>) return false
    return AbstractList.orderedEquals(this, other)
}

override fun hashCode(): Int = AbstractList.orderedHashCode(this)

override fun checkIsMutable(): Unit = this@AbstractMutableMap.checkIsMutable()
}

actual override fun remove(key: K): V? {
    checkIsMutable()
    val iter = entries.iterator()
    while (iter.hasNext()) {
        val entry = iter.next()
        val k = entry.key
        if (key == k) {
            val value = entry.value
            iter.remove()
            return value
        }
    }
    return null
}

/**
 * This method is called every time when a mutating method is called on this mutable map.
 * Mutal maps that are built (frozen) must throw `UnsupportedOperationException`.
 * *ann internal open fun checkIsMutable(): Unit { */
/* *
 * * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * * Use of this source code is governed by the Apache 2.0 license that can be found in the
 * license/LICENSE.txt file.
 */
package kotlin.collections

/**
 * Provides a skeletal implementation of the
 * [MutableSet] interface.
 */
public actual abstract class AbstractMutableSet<E> protected actual constructor() :
AbstractMutableCollection<E>(), MutableSet<E> {

/**
 * Compares this set with another set instance with
 * unordered structural equality.
 * *ann * @return `true`, if [other] instance is a `Set` of the same size, all
 * elements of which are contained in this set.
 */
override fun equals(other: Any?): Boolean {
    if (other === this) return true
    if (other !is Set<*>) return false
    return AbstractSet.setEquals(this, other)
}

/**
 * Returns the hash code value for this set.
 */
override fun hashCode(): Int = AbstractSet.unorderedHashCode(this)
}

package kotlin.collections

/**
 * Provides a [MutableList] implementation,
 * which uses a resizable array as its backing storage.
 * This implementation doesn't provide a way to manage
 * capacity, as backing JS array is resizeable itself.
 * There is no speed advantage to pre-allocating array sizes in
 * JavaScript, so this implementation does not include any of the
 * * and \"growth increment\" concepts.
 * *anonymous actual open class ArrayList<E>: internal constructor(private var array: Array<E> = ArrayList())

/**
 * Creates an empty [ArrayList].
 */
public actual constructor() : this(0)

/**
 * Creates an empty [ArrayList].
 * @param initialCapacity initial capacity (ignored)
 */
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual constructor(initialCapacity: Int = 0) : this(0)

/**
 * Creates an empty [ArrayList] filled from the
 * [elements] collection.
 */
public actual constructor(elements: Collection<E>) : this(elements.toTypedArray())

@Suppress("UNCHECKED_CAST")
actual override val size: Int get() = array.size

@Suppress("UNCHECKED_CAST")
actual override fun add(element: E): Boolean {
    checkIsMutable()
    array.asDynamic().push(element)
}
modCount++

modCount++

actual override fun add(index: Int, element: E): Unit {

checkIsMutable()

array.asDynamic().splice(insertionRangeCheck(index), 0, element)

modCount++

actual override fun addAll(elements: Collection<E>): Boolean {

checkIsMutable()

insertionRangeCheck(index)

if (elements.isEmpty()) return false

array += elements.toTypedArray<Any?>() + array

else

array.asDynamic().splice(index, 1).un shifting

return true

}
start: Int, end: Int, comparator: Comparator<in T>): Array<T> { if (start == end) return array
val median = (start + end) / 2
val left = mergeSort(array, buffer, start, median, comparator)
val right = mergeSort(array, buffer, median + 1, end, comparator)
val target = if (left == buffer) array else buffer
// Merge
var leftIndex = start
val rightIndex = median + 1
for (i in start..end) {
    when {
        leftIndex <= median && rightIndex <= end -> {
            target[i] = leftValue
            leftIndex++
        }
        leftIndex <= median -> {
            target[i] = leftValue
            leftIndex++
        }
        else /* rightIndex <= end */ -> {
            target[i] = rightValue
            rightIndex++
        }
    }
}
return target
}
* @param loadFactor the load factor (ignored)\n * \n * @throws IllegalArgumentException if the initial capacity or load factor are negative\n * \n * @Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS") \n * \n actual constructor(initialCapacity: Int, loadFactor: Float = 0.0f) \n { \n // This implementation of HashMap has no need of load factors or \n capacities. require(initialCapacity >= 0) \n { "Negative initial capacity: $initialCapacity" } \n require(loadFactor >= 0) \n { "Non-positive load factor: $loadFactor" } this(initialCapacity, 0.0f) \n \n actual constructor(initialCapacity: Int) \n { \n || || \n actual constructor(original: Map<out K, V>) \n { \n this.putAll(original) \n \n actual override fun clear() \n \n internalMap.clear() // \n structureChanged(this) \n \n actual override fun containsKey(key: K): Boolean = \n internalMap.containsKey(key) \n \n actual override fun containsValue(value: V): Boolean = \n internalMap.any { \n equality.equals(it.value, value) } \n \n private var _entries: MutableSet<MutableMap.MutableEntry<K, V>>? = \n null \n \n actual override val entries: MutableSet<MutableMap.MutableEntry<K, V>> get() \n { \n if (_entries == null) { \n _entries = createEntrySet() \n } return _entries!! \n } \n \n protected open fun createEntrySet(): MutableSet<MutableMap.MutableEntry<K, V>> = EntrySet() \n \n actual override operator fun get(key: K): V? = \n internalMap.get(key) \n \n actual override fun put(key: K, value: V): V? = \n internalMap.put(key, value) \n \n actual override fun remove(key: K): V? = \n internalMap.remove(key) \n \n actual override val size: Int \n { \n Int get() = internalMap.size \n } \n \n actual override fun containsKey(key: K): Boolean = \n internalMap.containsKey(key) \n \n actual override fun containsValue(value: V): Boolean = \n internalMap.any { \n equality.equals(it.value, value) } \n \n private var _entries: MutableSet<MutableMap.MutableEntry<K, V>>? = \n null \n \n actual override val entries: MutableSet<MutableMap.MutableEntry<K, V>> get() \n { \n if (_entries == null) { \n _entries = createEntrySet() \n } return _entries!! \n } \n \n protected open fun createEntrySet(): MutableSet<MutableMap.MutableEntry<K, V>> = EntrySet() \n \n actual constructor(initialCapacity: Int, loadFactor: Float = 0.0f) \n { \n map = HashMap<E, Any>(initialCapacity, loadFactor) \n } \n \n actual constructor(initialCapacity: Int) \n { \n \n map = HashMap<E, Any>() \n } \n \n /* \n * Constructs a new empty [HashSet]. \n * \n * @param map underlying map to use. \n * \n */ \n \n actual constructor(elements: Collection<E>) \n { \n map = HashMap<E, Any>(elements.size) \n addAll(elements) \n } \n \n /* \n * Constructs a new empty [HashSet] \n * \n * @param initialCapacity the initial capacity \n * \n */ \n \n actual override fun containsKey(element: E): Boolean \n { \n val old = map.put(element, this) \n return old == null \n } \n \n actual override fun containsValue(value: V): Boolean \n { \n \n } \n \n actual override fun contains(element: E): Collection<E> \n { \n return map.containsKey(element) || \n containsValue(value) \n } \n \n actual override fun remove(element: E): Collection<E> \n { \n map.remove(element) \n \n } \n \n actual override fun clear() \n \n } \n \n /* \n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors. \n * Use of this source code is governed by the Apache 2.0 license that \n * can be found in the LICENSE.txt file. \n */
class InternalStringMap<K, V>(override val equality: EqualityComparator) : InternalMap<K, V> {

    private var backingMap: dynamic = createJsMap()
    override var size: Int = 0
        private set

    override operator fun contains(key: K): Boolean {
        if (key !is String) return false
        return backingMap[key] !== undefined
    }

    override operator fun get(key: K): V? {
        if (key !is String) return null
        val value = backingMap[key]
        return if (value !== undefined) value.unsafeCast<V>() else null
    }

    override fun put(key: K, value: V): V? {
        require(key is String)
        val oldValue = backingMap[key]
        backingMap[key] = value

        if (oldValue === undefined) {
            size++
            return null
        } else {
            return oldValue.unsafeCast<V>()
        }
    }

    override fun remove(key: K): V? {
        if (key !is String) return null
        val value = backingMap[key]
        if (value !== undefined) {
            jsDeleteProperty(backingMap, key)
            size--
            return value.unsafeCast<V>()
        } else {
            return null
        }
    }

    override fun clear() {
        backingMap = createJsMap()
        size = 0
    }

    override fun iterator(): MutableIterator<MutableEntry<K, V>> {
        return object : MutableIterator<MutableEntry<K, V>> {
            private val keys: Array<String> = js("Object").keys(backingMap)
            private val iterator = keys.iterator()
            private var lastKey: String? = null

            override fun hasNext(): Boolean = iterator.hasNext()

            override fun next(): MutableEntry<K, V> {
                val key = iterator.next()
                lastKey = key

                @Suppress("UNCHECKED_CAST")
                return newMapEntry(key as K)
            }

            override fun remove() {
                @Suppress("UNCHECKED_CAST")
                this@InternalStringMap.remove(lastKey!!)
            }
        }
    }

    private fun newMapEntry(key: K): MutableEntry<K, V> = object : MutableEntry<K, V> {
        override val key: K get() = key
        override val value: V get() = this@InternalStringMap[key].unsafeCast<V>()

        override fun override fun
    }
setValue(newValue: V): V = this@InternalStringMap.put(key, newValue).unsafeCast<V>()

override fun hashCode(): Int = AbstractMap.entryHashCode(this)
        override fun toString(): String = AbstractMap.entryToString(this)
        override fun equals(other: Any?): Boolean = AbstractMap.entryEquals(this, other)
/*
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

/*
 * Based on GWT LinkedHashMap
 */
package

/*
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

/*
 * Based on GWT LinkedHashMap
 */
package

/*
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

/*
 * Based on GWT LinkedHashMap
 */
package

/*
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
element, move head to next
head = next

next = null
prev = null
/* The hashmap that keeps track of our entries and
the chain. Note that we * duplicate the key here to eliminate changes to HashMap and minimize the code
here, at the expense of additional space.*/
private var map: HashMap<K, ChainEntry<K, V>>
private
var isReadOnly: Boolean = false

/* Constructs an empty [LinkedHashMap] instance. */
actual constructor() : super() {
map = HashMap<K, ChainEntry<K, V>>()
internal constructor(backingMap: HashMap<K, Any>): super() {
    @Suppress("UNCHECKED_CAST") // expected
to work due to erase
map = backingMap as HashMap<K, ChainEntry<K, V>>
/* Constructs an empty [LinkedHashMap] instance */
*
* @param initialCapacity the initial capacity
(ignored)
*
* @param loadFactor the load factor (ignored)
*
* @throws IllegalArgumentException if the initial capacity or load factor are negative
*
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
actual constructor(initialCapacity: Int, loadFactor: Float = 0.0f) : super()
    {
        map = HashMap<K, ChainEntry<K, V>>()
        this.putAll(initialMap)
        internal fun build(): HashMap<K, V> = EntrySet()
        actual override fun get(key: K): V? = map.get(key)
        actual override fun put(key: K, value: V): V? {
            checkIsMutable()
            val old = map.get(key)
            if (old == null) {
                val newEntry = ChainEntry(key, value)
                map.put(key, newEntry)
                newEntry.addToEnd()
                return null

            } else {
                return old.setValue(value)
            }
        }
        actual override fun remove(key: K): V? {
            checkIsMutable()
            val entry = map.remove(key)
            if (entry != null) {
                entry.remove()
                return entry.value
            } return null
        }
        actual override val size: Int get() = map.size
        internal override fun checkIsMutable() {
            if (isReadOnly) throw
                UnsupportedOperationException()
        }
    }

    override fun clone(): Any {
        return LinkedHashMap(this)
    }

    /*
     * Copyright 2010-2018
     * JetBrains s.r.o. and Kotlin Programming Language contributors.
     * Use of this source code is governed by the
     * Apache 2.0 license that can be found in the license/LICENSE.txt file.
     */
    /*
     * Based on GWT LinkedHashSet
     * Copyright 2008 Google Inc.
     */
    package kotlin.collections
    public actual open class LinkedHashSet<E> : HashSet<E>, MutableSet<E> {
        internal constructor(map: LinkedHashMap<E, Any>) : super(map)
        actual constructor() : super(LinkedHashMap<E, Any>)
        /* Constructs a new empty
         * [LinkedHashSet] filled with the elements of the specified collection. */
        actual constructor(elements: Collection<E>) {
            addAll(elements)
        }
        actual constructor(initialCapacity: Int, loadFactor: Float = 0.0f) : super(LinkedHashMap<E, Any>?)
        {
            @Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
            actual constructor(initialCapacity: Int)
        }
    }
}
internal fun build(): Set<E> {
    (map as LinkedHashMap<E, Any>).build()
    return this
}

internal override fun checkIsMutable(): Unit = map.checkIsMutable()

public override fun clone(): Any {
//    return LinkedHashSet(this)
//}

/**
 * Creates a new instance of the specialized implementation of
 * [LinkedHashSet] with the specified [String] elements, which elements the keys as properties of JS object
 * without hashing them.
 */
public fun linkedStringSetOf(vararg elements: String): LinkedHashSet<String> {
    return LinkedHashSet(linkedStringMapOf<Any>()).apply { addAll(elements) }
}

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin

import kotlin.contracts.*

@kotlin.internal.InlineOnly
public actual inline fun <R> synchronized(lock: Any, block: () -> R): R {
    contract {
        callsInPlace(block, InvocationKind.EXACTLY_ONCE)
    }
    return block()
}

package kotlin.io

internal abstract class BaseOutput {
    open fun println() {
        print("\n")
    }

    open fun println(message: Any?) {
        print(message)
        println()
    }

    abstract fun print(message: Any?)

    open fun flush() {}
}

/** JsName used to make the declaration available outside of module to test it */
@JsName("NodeJsOutput")
internal class NodeJsOutput(val outputStream: dynamic) : BaseOutput() {
    override fun print(message: Any?) {
        // TODO: Using local variable because of bug in block decomposition
        val messageString = String(message)
        outputStream.write(messageString)
    }
}

/** JsName used to make the declaration available outside of module to test it */
@JsName("OutputToConsoleLog")
internal class OutputToConsoleLog : BaseOutput() {
    override fun println(message: Any?) {
        console.log(message)
    }

    override fun println() {
        console.log("")
    }
}

/** JsName used to make the declaration available outside of module to test it and use at try.kotl.in */
@JsName("BufferedOutput")
internal open class BufferedOutput : BaseOutput() {
    var buffer = ""

    override fun print(message: Any?) {
        buffer += String(message)
    }

    override fun flush() {
        buffer = ""
    }
}

/** JsName used to make the declaration available outside of module to test it */
@JsName("BufferedOutputToConsoleLog")
internal class BufferedOutputToConsoleLog : BufferedOutput() {
    override fun println(message: Any?) {
        var s = String(message)
        val i = s.nativeLastIndexOf("\n", 0)
        if (i >= 0) {
            buffer += s.substring(0, i)
            flush()
            s = s.substring(i + 1)
        }
        buffer += s
    }
}

package kotlin.coroutines

import kotlin.coroutines.intrinsics.*
import kotlin.coroutines.intrinsics.COROUTINE_SUSPENDED

@PublishedApi
@SinceKotlin("1.3")
internal actual class SafeContinuation<in T>
internal actual constructor(
    delegate: Continuation<T>,
    initialResult: Any?) : Continuation<T> {
    @PublishedApi
    internal actual constructor(delegate: Continuation<T>, initialResult: Any?) : Continuation<T> {
        this.delegate = delegate
        this.delegateContext = private var result: Any = initialResult
    }

    override fun resumeWith(result: Result<T>) {
        this.result = result
    }
}

fun main() {
    println("Hello, World!")
}
delegate.resumeWith(result) | else -> throw
IllegalStateException("Already resumed") | throw
getOrThrow(): Any? | @PublishedApi
internal actual fun
gorun | result = COROUTINE_SUSPENDED
return COROUTINE_SUSPENDED | val result = this.result
return when | result ===
RESUMED - COROUTINE_SUSPENDED // already called continuation, indicate COROUTINE_SUSPENDED
upstream
result is Result.Failure -> throw result.exception
else -> result // either
COROUTINE_SUSPENDED or data
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resumed"
"Already resu...
is moved to another package, use 'kotlinx.dom.isText' instead.

replaceWith = ReplaceWith("this.isText", "kotlinx.dom.isText")

@DeprecatedSinceKotlin(warningSince = "1.4")
public val Node.isText: Boolean
    inline get() = this.newIsText()

/**
 * Gets a value indicating whether this node is an [Element].
 */
@LowPriorityInOverloadResolution
@Deprecated(
    message = "This API is moved to another package,
    use 'kotlinx.dom.isElement' instead.",
    replaceWith = ReplaceWith("this.isElement", "kotlinx.dom.isElement")
)
@DeprecatedSinceKotlin(warningSince = "1.4")
public val Node.isElement: Boolean
    inline get() = this.newIsElement()

/**
 * Returns the view of this `ItemArrayLike<T>` collection as `List<T>`
 */
public fun <T> ItemArrayLike<T>.asList(): List<T> = object : AbstractList<T>() {
    override val size: Int get() = this@asList.length

    override fun get(index: Int): T = when (index) {
        in 0..lastIndex ->
            this@asList.item(index).unsafeCast<T>()
        else -> throw IndexOutOfBoundsException("index $index is not in range [0..$lastIndex]")
    }
}

/**
 * Creates text node and append it to the element.
 * 
 * @return this element
 */
@LowPriorityInOverloadResolution
@Deprecated(
    message = "This API is moved to another package,
    use 'kotlinx.dom.appendText' instead.",
    replaceWith = ReplaceWith("this.appendText(text)", "kotlinx.dom.appendText")
)
@DeprecatedSinceKotlin(warningSince = "1.4")
inline fun Element.appendText(text: String): Element = this.newAppendText(text)

/**
 * Allows to iterate this `dynamic` object in the following cases:
 * - when it has an `iterator` function,
 * - when it is an array
 * - when it is an instance of [kotlin.collections.Iterable]
 */
dynamic.iterator(): Iterator<dynamic> {
    val r: Any? = this

    return when {
        this["iterator"] != null ->
            this["iterator"]()
        isArrayish(r) ->
            r.unsafeCast<Array<*>>().iterator().unsafeCast()<T>()
        else ->
            r as Iterable<*>.iterator().unsafeCast()<T>()
    }
}

/**
 * Reinterprets this value as a value of the [dynamic type](/docs/reference/dynamic-type.html).
 */
der dynamic.unsafeCast(): @kotlin.internal.NoInfer T = this

dynamic.unsafeCastAny(): Any = dynamic.unsafeCastDynamic()

dynamic.unsafeCastDynamic(): dynamic = this

dynamic.unsafeCastDynamicAny(): dynamic = this

@JsName("unsafeCastDynamic")
dynamic.unsafeCastAny(): Any = dynamic.unsafeCastDynamic()

dynamic.unsafeCastDynamic(): dynamic = this

dynamic.unsafeCastDynamicAny(): dynamic = this

@kotlin.internal.InlineOnly
public inline fun <T> Any?.asDynamic(): dynamic = this

@kotlin.internal.InlineOnly
public inline fun <T> Any?.unsafeCast(): @kotlin.internal.NoInfer T = this.asDynamic()
module

@JsName("throwNPE")
internal fun throwNPE(message: String) {
    throw

}

@JsName("throwCCE")
internal fun throwCCE() {
    throw

}

@JsName("throwISE")
internal fun throwISE(message: String) {
    throw

}

@JsName("throwUPAE")
internal fun throwUPAE(propertyName: String) {
    throw

}

@file:kotlin.jvm.JvmName("GroupingKt")
@file:kotlin.jvm.JvmMultifileClass

package

/**
 * Represents a source of elements with a [keyOf] function, which can be applied to each element to get its key. 
 * A [Grouping] structure serves as an intermediate step in group-and-fold operations:
 * they group elements by their keys and then fold each group with some aggregating operation.
 * It is created by attaching `keySelector: (T) -> K` function to a source of elements.
 * To get an instance of [Grouping] use one of `groupingBy` extension functions:
 * - [Iterable.groupingBy]
 * - [Sequence.groupingBy]
 * - [Array.groupingBy]
 * - [CharSequence.groupingBy]
 * For the list of group-and-fold operations available, see the [extension functions](#extension-functions) for `Grouping`.
 * @sample samples.collections.Grouping.aggregateByRadix
 * Groups elements from the [Grouping] source by key and applies [operation] to the elements of each group sequentially.
 * @param operation a function that is invoked on each element with the following parameters:
 *  - `key`: the key of the group this element belongs to;
 *  - `accumulator`: the current value of the accumulator of the group, can be `null` if it's the first `element` encountered in the group;
 *  - `element`: the element from the source being aggregated;
 * @param first: indicates whether it's the first `element` encountered in the group.
 * @return the [destination] map associating the key of each group with the result of aggregation of the group elements.
 * @sample
 * Groups elements from the [Grouping] source by key and applies [operation] to the elements of each group sequentially.
 * @param operation a function that is invoked on each element with the following parameters:
 *  - `key`: the key of the group this element belongs to;
 *  - `accumulator`: the current value of the accumulator of the group, can be `null` if it's the first `element` encountered in the group;
 *  - `element`: the element from the source being aggregated;
 * @param first: indicates whether it's the first `element` encountered in the group.
 * @return the [destination] map associating the key of each group with the result of aggregation of the group elements.
 * @sample
 * Groups elements from the [Grouping] source by key and applies [operation] to the elements of each group sequentially.
 * @param operation a function that is invoked on each element with the following parameters:
 *  - `key`: the key of the group this element belongs to;
 *  - `accumulator`: the current value of the accumulator of the group, can be `null` if it's the first `element` encountered in the group;
 *  - `element`: the element from the source being aggregated;
 * @param first: indicates whether it's the first `element` encountered in the group.
 * @return the [destination] map associating the key of each group with the result of aggregation of the group elements.
 */

@SinceKotlin("1.1")
    M

@SinceKotlin("1.1")
public inline fun <T, K> Grouping<T, K>.eachCount(): Map<K, Int> =
    fold(0) { acc, _ -> acc + 1 }

@SinceKotlin("1.1")
public inline fun <T, K> Grouping<T, K>.eachSumOf(valueSelector: (T) -> Int): Map<K, Int> =
    fold(0) { acc, e -> acc + valueSelector(e) }

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 */

package kotlin.collections

/**
 * Groups elements from the [Grouping] source by key and counts elements in each group.
 * @return a [Map] associating the key of each group with the count of element in the group.
 */
@SinceKotlin("1.1")
public actual fun <T, K> Grouping<T, K>.eachCount(): Map<K, Int> =
    fold(0) { acc, _ -> acc + 1 }

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 */

@file:kotlin.jvm.JvmName("GroupingKt")
@file:kotlin.jvm.JvmMultifileClass

package

/**
 * Groups elements from the [Grouping] source by key and applies [operation] to the elements of each group sequentially, 
 * passing the previously accumulated value and the current element as arguments, and stores the results in a new map.
 * The key for each element is provided by the [Grouping.keyOf] function.
 * @param operation a function that is invoked on each element with the following parameters:
 *  - `key`: the key of the group this element belongs to;
 *  - `accumulator`: the current value of the accumulator of the group, can be `null` if it's the first `element` encountered in the group;
 *  - `element`: the element from the source being aggregated;
 * @param first: indicates whether it's the first `element` encountered in the group.
 * @return the [destination] map associating the key of each group with the result of aggregation of the group elements.
 */

@SinceKotlin("1.1")
    M

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 */

@file:kotlin.jvm.JvmName("GroupingKt")
@file:kotlin.jvm.JvmMultifileClass

package

/**
 * Groups elements from the [Grouping] source by key and applies [operation] to the elements of each group sequentially, 
 * passing the previously accumulated value and the current element as arguments, and stores the results in a new map.
 * The key for each element is provided by the [Grouping.keyOf] function.
 * @param operation a function that is invoked on each element with the following parameters:
 *  - `key`: the key of the group this element belongs to;
 *  - `accumulator`: the current value of the accumulator of the group, can be `null` if it's the first `element` encountered in the group;
 *  - `element`: the element from the source being aggregated;
 * @param first: indicates whether it's the first `element` encountered in the group.
 * @return the [destination] map associating the key of each group with the result of aggregation of the group elements.
 */

@SinceKotlin("1.1")
    M

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 */
aggregateTo(destination: M, operation: (key: K, accumulator: R?, element: T, first: Boolean) -> R?) =

@Suppress("UNCHECKED_CAST")
    aggregateTo(destination) { key, acc, e, first -> operation(key, if (first) acc as R, e) }

* sample collections.Grouping.foldByEvenLengthWithComputedInitialValue

* sample collections.Grouping.foldByEvenLengthWithConstantInitialValue

* sample collections.Grouping.foldByEvenLengthWithInitialValueSelector

* sample collections.Grouping.foldByEvenLengthWithInitialValueSelectorTo

* sample collections.Grouping.foldByEvenLengthWithInitialValueTo

@sample
the accumulator of the group:

- `key`: the element from the source being accumulated.
- @return the [destination] map associating the key of each group with the result of accumulating the group elements.
- @sample samples.collections.Grouping.foldByEvenLengthWithConstantInitialValueTo

```
@SinceKotlin("1.1")
public inline fun <T, K, R, M : MutableMap<in K, R>> Grouping<T, K>.foldTo(
    destination: M,
    initialValue: R,
    operation: (accumulator: R, element: T) -> R
): M =
    aggregateTo(destination) { _, acc, e, first -> operation(accumulator, e) }
```

- `key`: the current value of the accumulator of the group;
- `value`: the element from the source being accumulated.
- @return a [Map] associating the key of each group with the result of accumulating the group elements.
- @sample samples.collections.Grouping.reduceByMaxVowels

```
@SinceKotlin("1.1")
public inline fun <S, T : S, K> Grouping<T, K>.reduce(
    operation: (key: K,
    accumulator: S, element: T) -> S
): Map<K, S> =
    aggregate { key, acc, e, first ->
        if (first) e else operation(key, acc as S, e)
    }
```

- `key`: the element from the source being accumulated;
- `value`: the first element of the group;
- @param operation a function that is invoked on each element of the group with the following parameters:
- @return the [destination] map associating the key of each group with the result of accumulating the group elements.
- @sample samples.collections.Grouping.reduceByMaxVowelsTo

```
@SinceKotlin("1.1")
public inline fun <S, T : S, K, M : MutableMap<in K, S>> Grouping<T, K>.reduceTo(
    destination: M,
    operation: (key: K,
    accumulator: S, element: T) -> S
): M =
    aggregateTo(destination) { key, acc, e, first ->
        if (first) e else operation(key, acc as S, e)
    }
```

- `key`: the element from the source being accumulated;
- `value`: the initial value of the accumulator for that group and the first element of that group is also skipped;
- @param operation a function that is invoked on each element of the group with the following parameters:
- @return the [destination] map associating the key of each group with the result of accumulating the group elements.
- @sample samples.collections.Grouping.eachCountTo

```
@SinceKotlin("1.1")
public fun <T, K, M : MutableMap<in K, Int>> Grouping<T, K>.eachCountTo(destination: M): M =
    foldTo(destination, 0) { acc, _ -> acc + 1 }
```

- `key`: the element from the source being accumulated;
- `value`: the initial value of the accumulator for that group and the first element of that group is also skipped;
- @param operation a function that is invoked on each element of the group with the following parameters:
- @return a [List] of the given [destination] map associating the key of each group with the result of accumulating the group elements.
- @sample samples.collections.Grouping.eachSumOfTo

```
@SinceKotlin("1.1")
    foldTo(destination, 0L) { acc, e -> acc + valueSelector(e) }
```

- `key`: the element from the source being accumulated;
- `value`: the initial value of the accumulator for that group and the first element of that group is also skipped;
- @param operation a function that is invoked on each element of the group with the following parameters:
- @return a [List] of the given [destination] map associating the key of each group with the result of accumulating the group elements.
- @sample samples.collections.Grouping.sumEachByLong

```
@SinceKotlin("1.1")
public inline fun <T, K, M : MutableMap<in K, Double>> Grouping<T, K>.sumEachByDoubleTo(destination: M, valueSelector: (T) -> Double): M =
    foldTo(destination, 0.0) { acc, e -> acc + valueSelector(e) }
```

- `key`: the element from the source being accumulated;
- `value`: the initial value of the accumulator for that group and the first element of that group is also skipped;
- @param operation a function that is invoked on each element of the group with the following parameters:
- @return a [List] of the given [destination] map associating the key of each group with the result of accumulating the group elements.

public inline fun <T, K> Grouping<T, K>.sumEachByDouble(valueSelector: (T) -> Double): Map<K, Double> =
        fold(0.0) { acc, e -> acc + valueSelector(e)}
**PublicApi**

```kotlin
@PublishedApi
    internal fun atanh(value: Double): Double

    @PublishedApi
    internal fun hypot(x: Double, y: Double): Double

    @PublishedApi
    internal fun log10(value: Double): Double

    @PublishedApi
    internal fun log2(value: Double): Double

    @PublishedApi
    internal fun log1p(value: Double): Double

    @PublishedApi
    internal fun expm1(value: Double): Double

    @PublishedApi
    internal fun clz32(value: Int): Int
```

```kotlin
/**
 * Returns the smaller of two values.
 */
@Suppress("DEPRECATION_ERROR")
@Deprecated("Use minOf or kotlin.math.min instead",
    ReplaceWith("minOf(a, b)"), level = DeprecationLevel.ERROR)
public fun Math.min(a: Long, b: Long): Long =
    if (a <= b) a else b

/**
 * Returns the greater of two values.
 */
@Suppress("DEPRECATION_ERROR")
@Deprecated("Use maxOf or kotlin.math.max instead",
    ReplaceWith("maxOf(a, b)"), level = DeprecationLevel.ERROR)
public fun Math.max(a: Long, b: Long): Long =
    if (a >= b) a else b
```

```kotlin
package kotlin.js

/**
 * An interface for indexing access to a collection of key-value pairs, where type of key is [String] and type of value is [Any]?[Any]. */
public external interface Json {
    /**
     * Calls to the function will be translated to indexing operation (square brackets) on the receiver with [propertyName] as the argument.
     */
    operator fun get(propertyName: String): Any?

    /**
     * Calls of the function will be translated to an assignment of [value] to the receiver indexed (with square brackets/index operation) with [propertyName].
     */
    operator fun set(propertyName: String, value: Any?): Unit
}

/**
 * Returns a simple JavaScript object (as [Json]) using provided key-value pairs as names and values of its properties.
 */
public fun json(vararg pairs: Pair<String, Any?>): Json {
    val res: dynamic = js("({})")
    for ((name, value) in pairs) {
        res[name] = value
    }
    return res
}

/**
 * Adds key-value pairs from [other] to [this].
 * Returns the original receiver.
 */
public fun Json.add(other: Json): Json {
    val keys: Array<String> =
        js("Object").keys(other)
    for (key in keys) {
        if (other.asDynamic().hasOwnProperty(key)) {
            this[key] = other[key];
        }
    }
    return this
}

/**
 */
@Suppress("NOT_DOCUMENTED")
public external object JSON {
    public fun stringify(o: Any?): String
    public fun stringify(o: Any?, replacer: ((key: String, value: Any?) -> Any?)?): String
    public fun stringify(o: Any?, replacer: ((key: String, value: Any?) -> Any?)? = definedExternally, space: Int): String
    public fun stringify(o: Any?, replacer: ((key: String, value: Any?) -> Any?)? = definedExternally, space: String): String
    public fun stringify(o: Any?, replacer: Array<String>): String
    public fun stringify(o: Any?, replacer: Array<String>, space: Int): String
    public fun stringify(o: Any?, replacer: Array<String>, space: String): String
}
```

---

@PublishedApi internal fun atanh(value: Double): Double

@PublishedApi internal fun hypot(x: Double, y: Double): Double

@PublishedApi internal fun log10(value: Double): Double

@PublishedApi internal fun log2(value: Double): Double

@PublishedApi internal fun log1p(value: Double): Double

@PublishedApi internal fun expm1(value: Double): Double

@PublishedApi internal fun clz32(value: Int): Int
```

```kotlin
/**
 * Returns the smaller of two values.
 */
@Suppress("DEPRECATION_ERROR")
@Deprecated("Use minOf or kotlin.math.min instead",
    ReplaceWith("minOf(a, b)"), level = DeprecationLevel.ERROR)
public fun Math.min(a: Long, b: Long): Long =
    if (a <= b) a else b

/**
 * Returns the greater of two values.
 */
@Suppress("DEPRECATION_ERROR")
@Deprecated("Use maxOf or kotlin.math.max instead",
    ReplaceWith("maxOf(a, b)"), level = DeprecationLevel.ERROR)
public fun Math.max(a: Long, b: Long): Long =
    if (a >= b) a else b
```

```kotlin
package kotlin.js

/**
 * An interface for indexing access to a collection of key-value pairs, where type of key is [String] and type of value is [Any]?[Any]. */
public external interface Json {
    /**
     * Calls to the function will be translated to indexing operation (square brackets) on the receiver with [propertyName] as the argument.
     */
    operator fun get(propertyName: String): Any?

    /**
     * Calls of the function will be translated to an assignment of [value] to the receiver indexed (with square brackets/index operation) with [propertyName].
     */
    operator fun set(propertyName: String, value: Any?): Unit
}

/**
 * Returns a simple JavaScript object (as [Json]) using provided key-value pairs as names and values of its properties.
 */
public fun json(vararg pairs: Pair<String, Any?>): Json {
    val res: dynamic = js("({})")
    for ((name, value) in pairs) {
        res[name] = value
    }
    return res
}

/**
 * Adds key-value pairs from [other] to [this].
 * Returns the original receiver.
 */
public fun Json.add(other: Json): Json {
    val keys: Array<String> =
        js("Object").keys(other)
    for (key in keys) {
        if (other.asDynamic().hasOwnProperty(key)) {
            this[key] = other[key];
        }
    }
    return this
}

/**
 */
@Suppress("NOT_DOCUMENTED")
public external object JSON {
    public fun stringify(o: Any?): String
    public fun stringify(o: Any?, replacer: ((key: String, value: Any?) -> Any?)?): String
    public fun stringify(o: Any?, replacer: ((key: String, value: Any?) -> Any?)? = definedExternally, space: Int): String
    public fun stringify(o: Any?, replacer: ((key: String, value: Any?) -> Any?)? = definedExternally, space: String): String
    public fun stringify(o: Any?, replacer: Array<String>): String
    public fun stringify(o: Any?, replacer: Array<String>, space: Int): String
    public fun stringify(o: Any?, replacer: Array<String>, space: String): String
}
```
Double = nativeMath.tan(x)/un/*"rn* Computes the arc sine of the value [x];\n* the returned value is an angle in the range from `-PI/2` to `PI/2` radians.El *n * Special cases:rn* - `asin(x)` is `NaN`, when `abs(x) > 1` or x is `NaN` *n@SinceKotlin("1.2")*/rn@InlineOnly\npublic actual inline fun asin(x: Double): Double = nativeMath.asin(x)/un/*"rn* Computes the arc cosine of the value [x];\n* the returned value is an angle in the range from `0.0` to `PI` radians.El *n * Special cases:rn* - `acos(x)` is `NaN`, when `abs(x) > 1` or x is `NaN` *n@SinceKotlin("1.2")*/rn@InlineOnly\npublic actual inline fun acos(x: Double): Double = nativeMath.acos(x)/un/*"rn* Computes the arc tangent of the value [x];\n* the returned value is an angle in the range from `-PI/2` to `PI/2` radians.El *n * Special cases:*n - `atan(NaN)` is `NaN` *n@SinceKotlin("1.2")*/rn@InlineOnly\npublic actual inline fun atan(x: Double): Double = nativeMath.atan(x)/un/*"rn* Computes the hyperbolic cosine of the value [x];\n* Special cases:*n - `cosh(NaN)` is `NaN` *n@SinceKotlin("1.2")*/rn@InlineOnly\npublic actual inline fun sinh(x: Double): Double = nativeMath.sinh(x)/un/*"rn* Computes the hyperbolic cosine of the value [x];\n* Special cases:*n - `cosh(NaN)` is `NaN` *n@SinceKotlin("1.2")*/rn@InlineOnly\npublic actual inline fun cosh(x: Double): Double = nativeMath.cosh(x)/un/*"rn* Computes the hyperbolic cosine of the value [x];\n* Special cases:*n - `cosh(NaN)` is `NaN` *n@SinceKotlin("1.2")*/rn@InlineOnly\npublic actual inline fun tanh(x: Double): Double = nativeMath.tanh(x)/un/*"rn* Computes the inverse hyperbolic cosine of the value [x];\n* Special cases:*n - `atanh(NaN)` is `NaN` *n@SinceKotlin("1.2")*/rn@InlineOnly\npublic actual inline fun exp(x: Double): Double = nativeMath.exp(x)/un/*"rn* Computes Euler's number `e` raised to the power of the value [x];\n* Special cases:*n - `exp(NaN)` is `NaN` *n@SinceKotlin("1.2")*/rn@InlineOnly\npublic actual inline fun expm1(x: Double): Double = nativeMath.expm1(x)/un/*"rn* Computes `exp(x) - 1` without intermediate overflow or underflow.*\n* Special cases:*n - `expm1(NaN)` is `NaN` *n@SinceKotlin("1.2")*/rn@InlineOnly\npublic actual inline fun sqrt(x: Double): Double = nativeMath.sqrt(x)/un/*"rn* Computes the square root of the value [x];\n* Special cases:*n - `sqrt(x)` is `NaN` when `x < 0` or `x` is `NaN` *n@SinceKotlin("1.2")*/rn@InlineOnly\npublic actual inline fun hypot(x: Double, y: Double): Double = nativeMath.hypot(x, y)/un/*"rn* Computes `sqrt(x^2 + y^2)` without intermediate overflow or underflow.*\n* Special cases:*n - `hypot(NaN, NaN)` is `NaN` *n@SinceKotlin("1.2")*/rn@InlineOnly\npublic actual inline fun log(x: Double): Double = nativeMath.log(x)/un/*"rn* Computes the logarithm of the value [x] to
the given [base].ln *n Special cases:
  * - `log(x, b)` is `NaN` if either `x` or `b` are `NaN`
  * - `log(0, b)` is `NaN` when `x < 0` or `b <= 0` or `b == 1.0`
  * - `log(x, +Inf)` is `-Inf` for `b > 1` and `-Inf` for `b < 1`
  * - `log(0.0, b)` is `-Inf` for `b > 1` and `+Inf` for `b < 1`

See also logarithm functions for common fixed bases: [ln], [log10] and [log2].

```kotlin
@SinceKotlin("1.2")
public actual fun log(x: Double, base: Double): Double =
    if (base <= 0.0 || base == 1.0) return Double.NaN
    return nativeMath.log(x) / nativeMath.log(base)
```

** Computes the natural logarithm (base `E`) of the value `[x]`. ln *n Special cases:
  * - `-ln(NaN)` is `NaN`
  * - `ln(x)` is `NaN` when `x < 0.0`
  * - `ln(+Inf)` is `+Inf`
  * - `ln(0.0)` is `-Inf`

```kotlin
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun ln(x: Double): Double =
    nativeMath.log(x)
```

** Computes the common logarithm (base 2) of the value `[x]`. ln *n @see [ln] function

```kotlin
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun log10(x: Double): Double =
    nativeMath.log10(x)
```

** Computes the binary logarithm (base 2) of the value `[x]`. ln *n @see [ln] function

```kotlin
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun log2(x: Double): Double =
    nativeMath.log2(x)
```

** Computes `ln(x + 1)`.

```kotlin
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun ln1p(x: Double): Double =
    nativeMath.log1p(x)
```

** Rounds the given value `[x]` to an integer towards positive infinity.

```kotlin
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun ceil(x: Double): Double =
    nativeMath.ceil(x).unsafeCast<Double>() // TODO: Remove unsafe cast after removing public js.math
```

** Rounds the given value `[x]` to an integer towards negative infinity.

```kotlin
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun floor(x: Double): Double =
    nativeMath.floor(x).unsafeCast<Double>()
```

** Returns the value `[x]` having its fractional part truncated.

```kotlin
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun truncate(x: Double): Double =
    nativeMath.trunc(x)
```

** Returns the closest integer with ties rounded towards even integer.

```kotlin
@SinceKotlin("1.2")
public actual fun round(x: Double): Double =
    if (x % 0.5 != 0.0) {
        return nativeMath.round(x).unsafeCast<Double>()
    }
    val floor = floor(x)
    return if (floor % 2 == 0.0) floor else ceil(x)
```

** Returns the absolute value of the given value `[x]`.

```kotlin
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun abs(x: Double): Double =
    nativeMath.abs(x)
```

** Returns the sign of the given value `[x]`:

```kotlin
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun sign(x: Double): Double =
    nativeMath.sign(x)
```

** Returns the smaller of two values.

```kotlin
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun min(a: Double, b: Double): Double =
    nativeMath.min(a, b)
```

** Returns the greater of two values.

```kotlin
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun max(a: Double, b: Double): Double =
    nativeMath.max(a, b)
```

** Returns the value `a` raised to the power of `b`.

```kotlin
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun pow(a: Double, b: Double): Double =
    nativeMath.pow(a, b)
```

** Returns the value `a` raised to the integer power `n`.

```kotlin
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun pow(n: Int): Double =
    nativeMath.pow(this, n)
```

Open Source Used In security_automation identity-guest-cs 1607
= nativeMath.pow(this, n.toDouble())

/**
 * Returns the absolute value of this value.
 */
@SinceKotlin("1.2")
@InlineOnly
public

/**
 * Returns the sign of this value:
 * - `1.0` if the value is positive,
 * - `-1.0` if the value is negative,
 * - `0` if the value is zero.
 */
@SinceKotlin("1.2")
@InlineOnly
public

/**
 * Returns this value with the sign bit same as of the [sign] value.
 */
@SinceKotlin("1.2")
@InlineOnly
public

/**
 * Returns the ulp (unit in the last place) of this value.
 */
@SinceKotlin("1.2")

/**
 * Returns the [Double] value nearest to this value in direction of positive infinity.
 */
@SinceKotlin("1.2")

/**
 * Rounds this [Double] value to the nearest integer and converts the result to [Int].
 */
@SinceKotlin("1.2")

/**
 * Returns the [Double] value nearest to this value in direction of negative infinity.
 */
@SinceKotlin("1.2")

/** Computes the sine of the angle [x] given in radians.
 */
@SinceKotlin("1.2")
@InlineOnly

/** Computes the cosine of the angle [x] given in radians.
 */
@SinceKotlin("1.2")
@InlineOnly

/** Computes the tangent of the angle [x] given in radians.
 */
@SinceKotlin("1.2")
@InlineOnly

/** Computes the arc sine of the value [x] when `abs(x)` is `NaN`, when `abs(x)` >
open source used in security automation identity-guest-cs 1609
Open Source Used In Security Automation Identity-Guest-Cs 1610

```kotlin
nativeMath.expm1(x.toDouble()).toFloat() // Computes the natural logarithm (base 'E') of the value [x].

/**
 * Computes the logarithm of the value [x] to the given [base].
 * Special cases:
 *   - log(x, b) = NaN when 'x' < 0 or 'b' <= 0 or 'x' == 1.0
 *   - log(+Inf, b) = NaN when 'b' = 1
 *   - log(0.0, b) = -Inf for 'b' > 1 and 'Inf' for 'b' < 1
 *   - log(0, b) is NaN for 'b' < 1
 *
 * @param x the value
 * @param b the base
 * @return the natural logarithm of [x]
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun log(x: Float, base: Float): Float = nativeMath.log(x.toDouble(), base.toDouble()).toFloat()

/**
 * Computes the natural logarithm
 * of the value [x].
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun log10(x: Float): Float = nativeMath.log10(x.toDouble()).toFloat()

/**
 * Computes the binary logarithm (base 2) of the value [x].
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun log2(x: Float): Float = nativeMath.log2(x.toDouble()).toFloat()
```

```kotlin
/**
 * Computes `ln1p(a)`.
 *
 * This function can be implemented to produce more precise result for [x] near zero.
 * Special cases:
 *   - ln1p(NaN) is NaN
 *   - ln1p(-1.0) is -Inf
 *   - ln1p(+1.0) is +Inf
 *   - ln1p(x) is NaN where 'x' < -1.0
 *
 * @see log extension function for special cases.
 * @see [expm1] function
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun ln1p(x: Float): Float = nativeMath.log1p(x.toDouble()).toFloat()
```

```kotlin
/**
 * Returns the absolute value of the given value [x].
 *
 * Special cases:
 *   - abs(NaN) is NaN
 *   - abs(+Inf) = +Inf
 *   - abs(-Inf) = -Inf
 *   - abs(x) is NaN when 'x' < 0
 *
 * Returns the absolute value [x] to an integer towards positive infinity.
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun abs(x: Float): Float = nativeMath.abs(x.toDouble()).toFloat()
```

```kotlin
/**
 * Returns the sign of the given value [x]:
 *   - 1.0 if the value is positive,
 *   - 0 if the value is zero,
 *   - -1.0 if the value is negative.
 *
 * For values that are not finite, the result is NaN.
 * Special cases:
 *   - sign(NaN) is NaN
 *   - sign(+Inf) = 1.0
 *   - sign(-Inf) = -1.0
 *
 * Returns the sign of the given value [x] to an integer towards zero.
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun sign(x: Float): Float = nativeMath.sign(x.toDouble()).toFloat()
```

```kotlin
/**
 * Rounds the given value [x] to an integer towards positive infinity.
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun ceil(x: Float): Float = nativeMath.ceil(x.toDouble()).toFloat()
```

```kotlin
/**
 * Rounds the given value [x] to an integer towards negative infinity.
 * Special cases:
 *   - floor(x) is x where 'x' is NaN or +Inf or -Inf or already a mathematical integer.
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun floor(x: Float): Float = nativeMath.floor(x.toDouble()).toFloat()
```

```kotlin
/**
 * Rounds the given value [x] towards the closest integer with ties rounded towards even integer.
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun truncate(x: Float): Float = truncate(x.toDouble()).toFloat()
```

```kotlin
/**
 * Rounds the given value [x] towards the closest integer with ties rounded towards even integer.
 * Special cases:
 *   - round(x) is x where 'x' is NaN or +Inf or -Inf or already a mathematical integer.
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun round(x: Float): Float = round(x.toDouble()).toFloat()
```

```kotlin
/**
 * Returns the smaller of two values.
 *
 * If either value is NaN, then the result is NaN.
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun min(a: Float, b: Float): Float = nativeMath.min(a.toDouble(), b.toDouble()).toFloat()
```

```kotlin
/**
 * Returns the greater of two values.
 *
 * If either value is NaN, then the result is NaN.
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun max(a: Float, b: Float): Float = nativeMath.max(a.toDouble(), b.toDouble()).toFloat()
```
can be found in the license/LICENSE.txt file.

```kotlin
/**
 * Returns `true` if the specified number is NaN value, `false` otherwise.
 */
public actual fun Double.isNaN(): Boolean = this != this

/**
 * Returns `true` if the argument is a finite floating-point value; returns `false` otherwise.
 */
public actual fun Float.isNaN(): Boolean = this != this

/**
 * Returns `true` if this value is infinitely large in magnitude.
 */
public actual fun Double.isInfinite(): Boolean = this == Double.POSITIVE_INFINITY || this == Double.NEGATIVE_INFINITY

/**
 * Counts the number of set bits in the binary representation of this [Int] number.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public actual fun Int.countOneBits(): Int {
    // Hacker's Delight 5-1 algorithm
    var v = this
    v = (v and 0x55555555) + (v.ushr(1) and 0x55555555)
    v = (v and 0x33333333) + (v.ushr(2) and 0x33333333)
    v = (v and 0x0F0F0F0F) + (v.ushr(4) and 0x0F0F0F0F)
    v = (v and 0x00FF00FF) + (v.ushr(8) and 0x00FF00FF)
    v = (v and 0x0000FFFF) + (v.ushr(16))
    return v
}

/**
 * Counts the number of consecutive most significant bits that are zero in the binary representation of this [Int] number.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@Suppress("DEPRECATION_ERROR")
@kotlin.internal.InlineOnly
public actual inline fun Int.countLeadingZeroBits(): Int =
    // Hacker's Delight 5-4 algorithm for expressing countTrailingZeroBits with countLeadingZeroBits
    Int.SIZE_BITS - (this or -this).inv().countLeadingZeroBits()

/**
 * Returns a number having a single bit set in the position of the least significant set bit of this [Int] number, or zero, if this number is zero.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public actual fun Int.takeLowestOneBit(): Int =
    // Hacker's Delight 2-1 algorithm for isolating rightmost 1-bit
    this and -this

/**
 * Rotates the binary representation of this [Int] number left by the specified [bitCount] number of bits.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public actual fun Int.rotateLeft(bitCount: Int): Int =
    shl(bitCount) or ushr(Int.SIZE_BITS - bitCount)

/**
 * Rotates the binary representation of this [Int] number right by the specified [bitCount] number of bits.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public actual fun Int.rotateRight(bitCount: Int): Int =
    shl(Int.SIZE_BITS - bitCount) or ushr(bitCount)

/**
 * Counts the number of set bits in the binary representation of this [Long] number.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public actual fun Long.countOneBits(): Long {
    // Hacker's Delight 5-1 algorithm
    var v = this
    v = (v and 0x5555555555555555L) + (v.ushr(1) and 0x5555555555555555L)
    v = (v and 0x3333333333333333L) + (v.ushr(2) and 0x3333333333333333L)
    v = (v and 0x0F0F0F0F0F0F0F0FL) + (v.ushr(4) and 0x0F0F0F0F0F0F0F0F)
    v = (v and 0x00FF00FF00FF00FF) + (v.ushr(8) and 0x00FF00FF00FF00FF)
    v = (v and 0x0000FFFF0000FFFF) + (v.ushr(16))
    return v
}

/**
 * Counts the number of consecutive most significant bits that are zero in the binary representation of this [Long] number.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@Suppress("DEPRECATION_ERROR")
@kotlin.internal.InlineOnly
public actual inline fun Long.countLeadingZeroBits(): Long =
    // Hacker's Delight 5-4 algorithm for expressing countTrailingZeroBits with countLeadingZeroBits
    Int.SIZE_BITS - (this or -this).inv().countLeadingZeroBits()

/**
 * Returns a number having a single bit set in the position of the most significant set bit of this [Long] number, or zero, if this number is zero.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public actual fun Long.takeHighestOneBit(): Long =
    // Hacker's Delight 2-1 algorithm for isolating leftmost 1-bit
    this and -this

/**
 * Rotates the binary representation of this [Long] number left by the specified [bitCount] number of bits.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public actual fun Long.rotateLeft(bitCount: Int): Long =
    shl(bitCount) or ushr(Int.SIZE_BITS - bitCount)

/**
 * Rotates the binary representation of this [Long] number right by the specified [bitCount] number of bits.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public actual fun Long.rotateRight(bitCount: Int): Long =
    shl(Int.SIZE_BITS - bitCount) or ushr(bitCount)
```

Open Source Used in security_automation identity-guest-cs 1612
Long.countOneBits(): Int =\n    high.countOneBits() + low.countOneBits() /* Counts the number of consecutive most significant bits that are zero in the binary representation of this [Long] number.*/

Long.countLeadingZeroBits(): Int =\n    when (val high = this.high) {\n        0 -> Int.SIZE_BITS +
        high.countLeadingZeroBits()\n    } else -> high.countLeadingZeroBits() /* Counts the number of consecutive least significant bits that are zero in the binary representation of this [Long] number.*/

Long.countTrailingZeroBits(): Int =\n    when (val low = this.low) {\n        0 -> Int.SIZE_BITS +
        low.countTrailingZeroBits()\n    } else -> low.countTrailingZeroBits() /* Returns a number having a single bit set in the position of the most significant bit of this [Long] number, * or zero, if this number is zero.*/

Long.takeHighestOneBit(): Long =\n    when (val high = this.high) {\n        0 -> Long(low.takeHighestOneBit(), 0)\n        else -> Long(0, high.takeHighestOneBit())\n    } /* Returns a number having a single bit set in the position of the least significant bit of this [Long] number, * or zero, if this number is zero.*/

Long.takeLowestOneBit(): Long =\n    when (val low = this.low) {\n        0 -> Long(0, high.takeLowestOneBit())\n        else -> Long(low, takeLowestOneBit())\n    } /* Returns the number left by a negative bit count as the same as rotating it right by the negated bit count:*/

Long.rotateLeft(bitCount: Int): Long {\n    if ((bitCount and 31) != 0) {\n        val low = this.low\n        val high = this.high\n        return if ((bitCount and 32) == 0) Long(low.shl(bitCount) or high.ushr(-bitCount), 0)\n            else Long(high, low.shl(bitCount))\n    } else {\n        return if ((bitCount and 32) == 0) this\n            else Long(high, low\n        }\n    } /* Rotates the binary representation of this [Long] number left by the specified [bitCount] number of bits.\n
Long.rotateRight(bitCount: Int): Long = rotateLeft(-bitCount) /* Rotates the binary representation of this [Long] number right by the specified [bitCount] number of bits.\n
Promise<out T>(executor: (resolve: (T) -> Unit, reject: (Throwable) -> Unit) -> Unit) {\n    public open fun <S> then(onFulfilled: ((T) -> S)?): Promise<S>\n    public open fun <S> then(onFulfilled: ((T) -> S)?, onRejected: ((Throwable) -> S)?): Promise<S>\n    public open fun <S> catch(onRejected: (Throwable) -> S): Promise<S>\n    companion object {\n        public fun <S> all(promise: Array<out Promise<S>>): Promise<Array<out S>>\n        public fun <S> race(promise: Array<out Promise<S>>): Promise<S>\n        public fun <S> reject(e: Throwable): Promise<S>\n        public fun <S> resolve(e: (S): Promise<S>\n    }

Promise<out T> then(\n    companion object {\n        public fun <S> all(promise: Array<out Promise<S>>): Promise<Array<out S>> {\n            Promise<Array<out S>>\n        }
isInstance(value: Any?): Boolean {
    return isInstanceFunction(value) // override val isInstance is defined in the given class
}

NothingKClassImpl : KClassImpl<Nothing> {
    override fun isInstance(value: Any?): Boolean = false
}

JsClass<Nothing> get() = throw UnsupportedOperationException("There's no native class for Nothing type")

override fun equals(other: Any?): Boolean = other == this && super.equals(other) // override val equals is defined in the given class

ErrorKClass : KClass<Nothing> {
    override fun isInstance(value: Any?): Boolean = error("Can't check isInstance on ErrorKClass")
}

NothingKClassImpl : KClassImpl<Nothing>(js("Object")) {
    override val simpleName: String = "Nothing"

    override fun isInstance(value: Any?): Boolean = false

    override fun equals(other: Any?): Boolean = other == this

    override val jClass: JsClass<Nothing> = throw UnsupportedOperationException("There's no native JS class for Nothing type")

    override fun hashCode(): Int = 0
}

internal class ErrorKClass : KClass<Nothing> {
    override val simpleName: String? = error("Unknown simpleName for ErrorKClass")
    override val qualifiedName: String? = error("Unknown qualifiedName for ErrorKClass")

    override fun isInstance(value: Any?): Boolean = error("Can't check isInstance on ErrorKClass")

    override fun equals(other: Any?): Boolean = other == this

    override fun hashCode(): Int = 0
}

internal class ErrorKClass : KClass<Nothing> {
    override val simpleName: String? = error("Unknown simpleName for ErrorKClass")
    override val qualifiedName: String? = error("Unknown qualifiedName for ErrorKClass")

    override fun isInstance(value: Any?): Boolean = error("Can't check isInstance on ErrorKClass")

    override fun equals(other: Any?): Boolean = other == this

    override fun hashCode(): Int = 0
}
PrivateKClassImpl(js("Function")), unsafeCast<JsClass<Any>>(js("Function")), length == arity})
  functionClasses.asDynamic()[arity] = result
  result
}
}

private val functionClasses =
  arrayOfNulls<KClassImpl<Any>>(0),

import kotlin.reflect.*
import kotlin.reflect.js.internal.*

@JsName("getKClass")
internal fun <T : Any>
  getKClass(jClass: Any /* JsClass<T> | Array<JsClass<T>> */): KClass<T> {
    return if
      (js("Array").isArray(jClass)) {
        getKClassM(jClass.unsafeCast<Array<JsClass<T>>>())
      } else {
        getKClass1(jClass.unsafeCast<JsClass<T>>())
      }
  }

@JsName("getKClassM")
internal fun <T : Any>
  getKClassM(jClasses: Array<JsClass<T>>): KClass<T> = when (jClasses.size) {
    1 ->
      getKClass1(jClasses[0])
    0 -> NothingKClassImpl.unsafeCast<KClass<T>>()
    else ->
      ErrorKClass().unsafeCast<KClass<T>>()
  }

@JsName("getKClassFromExpression")
internal fun <T : Any>
  getKClassFromExpression(e: T): KClass<T> =
    when (jsTypeOf(e)) {
      "string" -> PrimitiveClasses.stringClass
      "number" -> if (jsBitwiseOr(e, 0).asDynamic() === e)
        PrimitiveClasses.intClass
      else
        PrimitiveClasses.doubleClass
      "boolean" -> PrimitiveClasses.booleanClass
      "function" -> PrimitiveClasses.functionClass(e.asDynamic().length)
      else -> {
        when {
          e is BooleanArray -> PrimitiveClasses.booleanArrayClass
          e is CharArray -> PrimitiveClasses.charArrayClass
          e is ByteArray -> PrimitiveClasses.byteArrayClass
          e is ShortArray -> PrimitiveClasses.shortArrayClass
          e is IntArray -> PrimitiveClasses.intArrayClass
          e is LongArray -> PrimitiveClasses.longArrayClass
          e is FloatArray -> PrimitiveClasses.floatArrayClass
          e is DoubleArray -> PrimitiveClasses.doubleArrayClass
          else -> {
            val constructor = js("Object").getPrototypeOf(e).constructor
            when {
              constructor === js("Object") -> PrimitiveClasses.anyClass
              constructor === js("Error") -> PrimitiveClasses.throwableClass
              else -> {
                val jsClass: JsClass<T> = constructor
                getKClass1(jsClass)
              }
            }
          }
        }
      }
    }

@JsName("getKClass1")
internal fun <T : Any>
  getKClass1(jClass: JsClass<T>): KClass<T> {
    if (jClass === js("String")) return
      PrimitiveClasses.stringClass.unsafeCast<KClass<T>>()

    val metadata = jClass.asDynamic().`$metadata$`
    return if (metadata != null) {
      if (metadata.`$kClass$` == null) {
        val kClass =
          SimpleKClassImpl(jClass)
        metadata.`$kClass$` = kClass
        kClass
      } else {
        metadata.`$kClass$`
      }
    } else {
      SimpleKClassImpl(jClass)
    }
  }

package kotlin.js

/**
 * Exposes
 */
@Suppress("NOT_DOCUMENTED")
public external class RegExp(pattern: String, flags: String? = definedExternally) {

  public fun test(str: String): Boolean

  public fun exec(str: String): RegExpMatch?

  override fun toString(): String

  public val global: Boolean

  public val ignoreCase: Boolean

  public val multiline: Boolean

  /**
   * Resets the regular expression so that subsequent [RegExp.test] and [RegExp.exec] calls will match starting with the beginning of the input string, and causes the lastIndex property to be reset to 0.
   */
  public fun reset() {
    lastIndex = 0
  }

  // TODO: Inherit from array or introduce asArray() extension

  /**
   * Represents the return value of [RegExp.exec] calls with the beginning of the input string.
   */
  @Suppress("NOT_DOCUMENTED")
  public external interface RegExpMatch {
    public val index: Int

    public val input: String

    public val length: Int

    /**
     * Returns the entire text matched by [RegExp.exec] if the [index] parameter is 0, or the text matched by the capturing parenthesis at the given index.
     */
  }

  /**
   * The lastIndex is a read/write integer property of regular expressions that specifies the index at which to start the next match. Setting it to -1 has the same effect as calling [RegExp.reset].
   */

  public var lastIndex: Int

  /**
   * @returns the regular expression that is in effect when the [RegExp.exec] call was made. This method is not likely to be useful in most cases.
   */
  public fun get(): RegExp

  /**
   * If this regular expression object matches, call this method, passing in the capture groups in which the text was matched, to get the correct return values.
   */
  public fun getCaptureGroups(): Array<*>

/**
 * Converts the result of
 * [RegExp.exec] to an array where the first element contains the entire matched text and each subsequent
 * element is the text matched by each capturing parenthesis.
 */

public inline fun RegExpMatch.get(index: Int): String? = asDynamic()[index]

/**
 * Converts the result of
 * [RegExp.exec] to an array where the first element contains the entire matched text and each subsequent
 * element is the text matched by each capturing parenthesis.
 */

public inline fun RegExpMatch.get(index: Int): String? = asDynamic()[index]

/**
 * Converts the result of
 * [RegExp.exec] to an array where the first element contains the entire matched text and each subsequent
 * element is the text matched by each capturing parenthesis.
 */

public inline fun RegExpMatch.get(index: Int): String? = asDynamic()[index]

/**
 * Converts the result of
 * [RegExp.exec] to an array where the first element contains the entire matched text and each subsequent
 * element is the text matched by each capturing parenthesis.
 */

public inline fun RegExpMatch.get(index: Int): String? = asDynamic()[index]

/**
 * Converts the result of
 * [RegExp.exec] to an array where the first element contains the entire matched text and each subsequent
 * element is the text matched by each capturing parenthesis.
 */

public inline fun RegExpMatch.get(index: Int): String? = asDynamic()[index]

/**
 * Converts the result of
 * [RegExp.exec] to an array where the first element contains the entire matched text and each subsequent
 * element is the text matched by each capturing parenthesis.
 */

public inline fun RegExpMatch.get(index: Int): String? = asDynamic()[index]
Open Source Used in security_automation identity-guest-cs 1620

AbstractList.checkPositionIndex(index, length)\nstring = string.substring(0, index) + value +
string.substring(index)\nreturn this\n/**\n* Inserts characters in the specified character array
[value] into this string builder at the specified [index] and returns this instance.\n*\n* The inserted characters go in the same order as in the [value] character array, starting at [index].\n*\n* @throws IndexOutOfBoundsException or [IllegalArgumentException] when [index] is less than zero or greater than the length of this string builder.\n*/
@SinceKotlin("1.4")\n@WasExperimental(ExperimentalStdlibApi::class)\nfun insert(index: Int, value: CharArray): StringBuilder =
StringBuilder(\nAbstractList.checkPositionIndex(index, length)\nstring = string.substring(0, index) + value.concatToString() +
string.substring(index)\nreturn this\n)**\n* Inserts characters in the specified character sequence [value] into this string builder at the specified [index] and returns this instance.\n*\n* The inserted characters go in the same order as in the [value] character sequence, starting at [index].\n*\n* @param index the position in this string builder to insert at.\n*\n* @throws IndexOutOfBoundsException or [IllegalArgumentException] when [index] is less than zero or greater than the length of this string builder.\n*/
@SinceKotlin("1.4")\n@WasExperimental(ExperimentalStdlibApi::class)\nfun insert(index: Int, value: Any?): StringBuilder =
StringBuilder(\nAbstractList.checkPositionIndex(index, length)\nstring = string.substring(0, index) + value.toString() + string.substring(index)\nreturn this\n)**\n* Inserts the string representation of the specified object [value] into this string builder at the specified [index] and returns this instance.\n*\n* @param index the position in this string builder to insert at.\n*\n* @param value the character sequence from which characters are inserted. If [value] is `null`, then the four characters `\"null\"` are inserted.\n*\n* @throws IndexOutOfBoundsException if [index] is less than zero or greater than the length of this string builder.\n*\n* @SinceKotlin("1.4")\n@WasExperimental(ExperimentalStdlibApi::class)\nfun insert(index: Int, value: CharSequence?): StringBuilder =
StringBuilder(\nAbstractList.checkPositionIndex(index, length)\nstring = string.substring(0, index) + value + string.substring(index)\nreturn this\n)**\n* Inserts characters in the specified character sequence [value] into this string builder at the specified [index] and returns this instance.\n*\n* The inserted characters go in the same order as in the [value] character sequence, starting at [index].\n*\n* @param index the position in this string builder to insert at.\n*\n* @param value the character sequence from which characters are inserted. If [value] is `null`, then the four characters `\"null\"` are inserted.\n*\n* @throws IndexOutOfBoundsException if [index] is less than zero or greater than the length of this string builder.\n*/
@SinceKotlin("1.4")\n@WasExperimental(ExperimentalStdlibApi::class)\nfun insert(index: Int, value: Char): StringBuilder =
StringBuilder(\nAbstractList.checkPositionIndex(index, length)\nstring = string.substring(0, index) + value + string.substring(index)\nreturn this\n)**\n* Inserts characters in the specified character sequence [value] into this string builder at the specified [index] and returns this instance.\n*\n* The inserted characters go in the same order as in the [value] character sequence, starting at [index].\n*\n* @param index the position in this string builder to insert at.\n*\n* @param value the character from which character is inserted. If [value] is `null`, then the four characters `\"null\"` are inserted.\n*\n* @throws IndexOutOfBoundsException if [index] is less than zero or greater than the length of this string builder.\n*/
@SinceKotlin("1.4")\n@WasExperimental(ExperimentalStdlibApi::class)\nfun insert(index: Int, value: String): StringBuilder =
StringBuilder(\nAbstractList.checkPositionIndex(index, length)\nval toInsert = value ?: \"null\"
if (toInsert is String) {\n\nstring = this.string.substring(0, index) + toInsert + this.string.substring(index)
return this
} else {
\nfor (i in length until newLength) {\n\nstring += \"u0000\"
}\n
string = string.substring(0, newLength)\n\n}
*\n* Returns a new [String] that contains characters in this string builder at [startIndex] (inclusive) and up to the [length] (exclusive).\n*\n* @throws IndexOutOfBoundsException if [startIndex] is less than zero or greater than the length of this string builder.\n*\n* @SinceKotlin("1.4")\n@WasExperimental(ExperimentalStdlibApi::class)\nfun substring(startIndex: Int): String =
StringBuilder(\nAbstractList.checkPositionIndex(startIndex, length)\nstring = string.substring(startIndex)\nreturn this\n)**\n* Returns a new [String] that contains characters in this string builder at [startIndex] (inclusive) and up to the [endIndex] (exclusive).\n*\n* @throws IndexOutOfBoundsException or [IllegalArgumentExce
[startIndex] or [endIndex] is out of range of this string builder indices or when `startIndex > endIndex`.

* @SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
actual fun substring(startIndex: Int, endIndex: Int): String { 
string = string.substring(startIndex, endIndex)
return this
}

/**
 * Substrings are copied into the destination string builder, 
 * if the backing storage of this string builder is larger than necessary to hold its current contents, 
 * then it may be resized to become more space efficient.
 * CALLING THIS METHOD MAY, BUT IS NOT REQUIRED TO, AFFECT THE VALUE OF THE [capacity] PROPERTY.
 * ATTEMPTS TO REDUCE STORAGE USED FOR THIS STRING BUILDER.
 * IF THE BACKING STORAGE OF THIS STRING BUILDER IS LARGER THAN NECESSARY TO HOLD ITS CURRENT CONTENTS,
 * THEN IT MAY BE RESIZED TO BECOME MORE SPACE EFFICIENT.
 * CALLING THIS METHOD MAY, BUT IS NOT REQUIRED TO, AFFECT THE VALUE OF THE [CAPACITY] PROPERTY.
 * ATTEMPTS TO REDUCE STORAGE USED FOR THIS STRING BUILDER.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
actual fun funToSize() { 

override fun toString(): String = string

/**
 * Copies characters from this string builder into the [destination] string builder.
 * If the backing storage of this string builder is larger than necessary to hold its current contents,
 * then it may be resized to become more space efficient.
 * CALLING THIS METHOD MAY, BUT IS NOT REQUIRED TO, AFFECT THE VALUE OF THE [CAPACITY] PROPERTY.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
actual fun trimToSize() { 

/**
 * Returns the character at the specified [index] from this string builder and returns this instance.
 * IF THE CHAR AT THE SPECIFIED [INDEX] IS PART OF A SUPPLEMENTARY CODE POINT, THIS METHOD DOES NOT REMOVE
 * THE ENTIRE SUPPLEMENTARY CHARACTER.
 * ATTEMPTS TO REDUCE STORAGE USED FOR THIS STRING BUILDER.
 * IF THE BACKING STORAGE OF THIS STRING BUILDER IS LARGER THAN NECESSARY TO HOLD ITS CURRENT CONTENTS,
 * THEN IT MAY BE RESIZED TO BECOME MORE SPACE EFFICIENT.
 * CALLING THIS METHOD MAY, BUT IS NOT REQUIRED TO, AFFECT THE VALUE OF THE [CAPACITY] PROPERTY.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
actual fun deleteRange(startIndex: Int, endIndex: Int): StringBuilder { 
  throw IndexOutOfBoundsException("startIndex: $startIndex, endIndex: $endIndex")
}

/**
 * Removes characters in the specified range from this string builder and returns this instance.
 * IF THE CHAR AT THE SPECIFIED [INDEX] IS PART OF A SUPPLEMENTARY CODE POINT, THIS METHOD DOES NOT REMOVE
 * THE ENTIRE SUPPLEMENTARY CHARACTER.
 * ATTEMPTS TO REDUCE STORAGE USED FOR THIS STRING BUILDER.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
actual fun deleteAt(index: Int): StringBuilder { 
  throw IndexOutOfBoundsException("startIndex: $start, endIndex: $end")
}

/**
 * Attempt to reduce storage used for this string builder.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
actual fun funToSize() { 

/**
 * Returns the character at the specified [index] from this string builder and returns this instance.
 * IF THE CHAR AT THE SPECIFIED [INDEX] IS PART OF A SUPPLEMENTARY CODE POINT, THIS METHOD DOES NOT REMOVE
 * THE ENTIRE SUPPLEMENTARY CHARACTER.
 * ATTEMPTS TO REDUCE STORAGE USED FOR THIS STRING BUILDER.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
actual fun deleteRange(startIndex: Int, endIndex: Int): String { 
  throw IndexOutOfBoundsException("startIndex: $start, endIndex: $end")
}

/**
 * Returns the character at the specified [index] from this string builder and returns this instance.
 * IF THE CHAR AT THE SPECIFIED [INDEX] IS PART OF A SUPPLEMENTARY CODE POINT, THIS METHOD DOES NOT REMOVE
 * THE ENTIRE SUPPLEMENTARY CHARACTER.
 * ATTEMPTS TO REDUCE STORAGE USED FOR THIS STRING BUILDER.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
actual fun deleteAt(index: Int): String { 
  throw IndexOutOfBoundsException("startIndex: $start, endIndex: $end")
}

/**
 * Returns the character at the specified [index] from this string builder and returns this instance.
 * IF THE CHAR AT THE SPECIFIED [INDEX] IS PART OF A SUPPLEMENTARY CODE POINT, THIS METHOD DOES NOT REMOVE
 * THE ENTIRE SUPPLEMENTARY CHARACTER.
 * ATTEMPTS TO REDUCE STORAGE USED FOR THIS STRING BUILDER.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
actual fun deleteRange(startIndex: Int, endIndex: Int): String { 
  throw IndexOutOfBoundsException("startIndex: $start, endIndex: $end")
}

/**
 * Returns the character at the specified [index] from this string builder and returns this instance.
 * IF THE CHAR AT THE SPECIFIED [INDEX] IS PART OF A SUPPLEMENTARY CODE POINT, THIS METHOD DOES NOT REMOVE
 * THE ENTIRE SUPPLEMENTARY CHARACTER.
 * ATTEMPTS TO REDUCE STORAGE USED FOR THIS STRING BUILDER.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
actual fun deleteAt(index: Int): String { 
  throw IndexOutOfBoundsException("startIndex: $start, endIndex: $end")
}
public fun toCharArray(destination: CharArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = this.length) {
    var dstIndex = destinationOffset
    for (index in startIndex until endIndex) {
        destination[dstIndex++] = string[index]
    }
}

public fun appendRange(value: CharArray, startIndex: Int, endIndex: Int): StringBuilder {
    @OptIn(ExperimentalStdlibApi::class)
    string += value.concatToString(startIndex, endIndex)
    return this
}

public fun appendRange(value: CharSequence, startIndex: Int, endIndex: Int): StringBuilder {
    val stringCsq = value.toString()
    string += stringCsq.substring(startIndex, endIndex)
    return this
}

public fun insertRange(index: Int, value: CharArray, startIndex: Int, endIndex: Int): StringBuilder {
    AbstractList.checkPositionIndex(index, this.length)
    string = string.substring(0, index) + value.concatToString(startIndex, endIndex) + string.substring(index)
    return this
}

public fun insertRange(index: Int, value: CharSequence, startIndex: Int, endIndex: Int): StringBuilder {
    AbstractList.checkPositionIndex(index, length)
    string = string.substring(0, index) + stringCsq.substring(startIndex, endIndex) + string.substring(index)
    return this
}

/**
 * Clears the content of this string builder making it empty and returns this instance.
 * @sample
 */
samples.text.Strings.clearStringBuilder

/*\n@SinceKotlin("1.3")\n@Suppress("EXTENSION_SHADOWED_BY_MEMBER", "NOTHING_TO_INLINE")\npublic actual inline fun StringBuilder.clear(): StringBuilder = this.clear()\n*\nSets the character at the specified [index] to the specified [value].\n* @throws IndexOutOfBoundsException if [index] is out of bounds of this string builder.*/
*\n@SinceKotlin("1.4")\n@WasExperimental(ExperimentalStdlibApi::class)\n@Suppress("EXTENSION_SHADOWED_BY_MEMBER", "NOTHING_TO_INLINE")\npublic actual inline fun StringBuilder.clear(): StringBuilder = this.clear()

/**
 * Sets the character at the specified [index] to the specified [value].
 * @throws IndexOutOfBoundsException if [index] is out of bounds of this string builder.\n*/
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@Suppress("EXTENSION_SHADOWED_BY_MEMBER", "NOTHING_TO_INLINE")
public actual inline operator fun StringBuilder.set(index: Int, value: Char) = this.set(index, value)

/**
 * Replaces characters in the specified range of this string builder with characters in the specified string [value] and returns this instance.\n * @param startIndex the beginning (inclusive) of the range to replace.\n * @param endIndex the end (exclusive) of the range to replace.\n * @param value the string to replace with.\n * @throws IndexOutOfBoundsException or [IllegalArgumentOutOfRangeException] if [startIndex] is less than zero, greater than the length of this string builder, or `startIndex > endIndex`.\n*/
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@Suppress("EXTENSION_SHADOWED_BY_MEMBER", "NOTHING_TO_INLINE")
public actual inline fun StringBuilder.setRange(startIndex: Int, endIndex: Int, value: String): StringBuilder =
    this.setRange(startIndex, endIndex, value)

/**
 * Removes the character at the specified [index] from this string builder and returns this instance.\n * If the `Char` at the specified [index] is part of a supplementary code point, this method does not remove the entire supplementary character.\n * @param index the index of `Char` to remove.\n * @throws IndexOutOfBoundsException if [index] is out of bounds of this string builder.\n*/
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@Suppress("EXTENSION_SHADOWED_BY_MEMBER", "NOTHING_TO_INLINE")
public actual inline fun StringBuilder.deleteAt(index: Int): StringBuilder = this.deleteAt(index)

/**
 * Removes characters in the specified range from this string builder and returns this instance.\n * @param startIndex the beginning (inclusive) of the range to remove.\n * @param endIndex the end (exclusive) of the range to remove.\n * @throws IndexOutOfBoundsException or [IllegalArgumentOutOfRangeException] when [startIndex] is out of range of this string builder indices or when `startIndex > endIndex`.\n*/
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@Suppress("EXTENSION_SHADOWED_BY_MEMBER", "NOTHING_TO_INLINE")
public actual inline fun StringBuilder.deleteRange(startIndex: Int, endIndex: Int): StringBuilder = this.deleteRange(startIndex, endIndex)

/**
 * Copies characters from this string builder into the [destination] character array.\n * @param destination the array to copy to.\n * @param destinationOffset the position in the array to copy to, 0 by default.\n * @param startIndex the beginning (inclusive) of the range to copy, 0 by default.\n * @param endIndex the end (exclusive) of the range to copy, length of this string builder by default.\n * @throws IndexOutOfBoundsException or [IllegalArgumentOutOfRangeException] when [startIndex] or [endIndex] is out of range of this string builder indices or when `startIndex > endIndex`.\n * @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset].\n * @param value the array from which characters are appended.
 * @param startIndex the beginning (inclusive) of the subarray to append.\n * @param endIndex the end (exclusive) of the subarray to append.\n * @param value the array from which characters are appended.\n * @param value the array from which characters are appended.\n * @throws IndexOutOfBoundsException or [IllegalArgumentOutOfRangeException] when [startIndex] or [endIndex] is out of range of the [value] array indices or when `startIndex > endIndex`.\n*/
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@Suppress("EXTENSION_SHADOWED_BY_MEMBER", "NOTHING_TO_INLINE")
public actual inline fun StringBuilder.toCharArray(destination: CharArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = this.length) =
    this.toCharArray(destination, destinationOffset, startIndex, endIndex)

/**
 * Appends characters in a subarray of the specified character array [value] to this string builder and returns this instance.\n * Characters are appended in order, starting at specified [startIndex].\n * @param value the array from which characters are appended.\n * @param startIndex the beginning (inclusive) of the subarray to append.\n * @param endIndex the end (exclusive) of the subarray to append.\n * @throws IndexOutOfBoundsException or [IllegalArgumentOutOfRangeException] when [startIndex] or [endIndex] is out of range of the [value] array indices or when `startIndex > endIndex`.\n*/
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@Suppress("EXTENSION_SHADOWED_BY_MEMBER", "NOTHING_TO_INLINE")
public actual inline fun StringBuilder.toCharArray(destination: CharArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = this.length) =
    this.toCharArray(destination, destinationOffset, startIndex, endIndex)
StringBuilder.appendRange(value: CharArray, startIndex: Int, endIndex: Int): StringBuilder =

StringBuilder.appendRange(value: CharSequence, startIndex: Int, endIndex: Int): StringBuilder =

StringBuilder.insertRange(index: Int, value: CharArray, startIndex: Int, endIndex: Int): StringBuilder =

StringBuilder.insertRange(index: Int, value: CharSequence, startIndex: Int, endIndex: Int): StringBuilder =

Char.isWhitespace(): Boolean = toString().matches("[\s\xA0]")

Char.toLowerCase(): Char = lowercaseChar()
Char.lowerCaseChar(): Char =
    toString().asDynamic().toLowerCase().charCodeAt(0).unsafeCast<Int>().toChar()

/**
 * Converts this character to lower case using Unicode mapping rules of the invariant locale.
 * This function supports one-to-many character mapping, thus the length of the returned string can be greater than one.
 * For example, \u0130.lowerCase() returns \u0069\u0307, where \u0130 is the LATIN CAPITAL LETTER I WITH DOT ABOVE character ('\u0130').
 * If this character has no lower case mapping, the result of toString() of this char is returned.
 */
@SinceKotlin("1.4")
@ExperimentalStdlibApi
public actual inline fun Char.lowerCase(): String = toString().asDynamic().toLowerCase() as String

/**
 * Converts this character to upper case using Unicode mapping rules of the invariant locale.
 * This function performs one-to-one character mapping.
 * To support one-to-many character mapping use the [lowerCase] function.
 * If this character has no mapping equivalent, the character itself is returned.
 */
@SinceKotlin("1.4")
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
public actual fun Char.upperCaseChar(): Char {
    val uppercase = upperCase()
    return if (uppercase.length > 1) this else uppercase[0]
}

/**
 * Converts this character to upper case using Unicode mapping rules of the invariant locale.
 * This function supports one-to-many character mapping, thus the length of the returned string can be greater than one.
 * For example, \uFB00.upperCase() returns \u0046\u0046, where \uFB00 is the LATIN SMALL LIGATURE FF character ('\uFB00').
 * If this character has no upper case mapping, the result of toString() of this char is returned.
 */
@SinceKotlin("1.4")
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
public actual inline fun Char.upperCase(): String = toString().asDynamic().toUpperCase() as String

/**
 * Returns `true` if this character is a Unicode high-surrogate code unit (also known as leading-surrogate code unit).
 */
public actual fun Char.isHighSurrogate(): Boolean = this in Char.MIN_HIGH_SURROGATE..Char.MAX_HIGH_SURROGATE

/**
 * Returns `true` if this character is a Unicode low-surrogate code unit (also known as trailing-surrogate code unit).
 */
public actual fun Char.isLowSurrogate(): Boolean = this in Char.MIN_LOW_SURROGATE..Char.MAX_LOW_SURROGATE

package kotlin.text

/**
 * Returns `true` if the content of this string is equal to the word "true", ignoring case, and `false` otherwise.
 */
@DeprecatedSinceKotlin(hiddenSince = "1.4")
public actual inline fun String.toBoolean(): Boolean = this.toBoolean()

/**
 * Parses the string as a signed [Byte] number and returns the result.
 * @throws NumberFormatException if the string is not a valid representation of a number.
 */
public actual fun String.toByte(): Byte = toByteOrNull() ?: numberFormatException(this)

/**
 * Parses the string as a signed [Byte] number and returns the result.
 * @throws NumberFormatException if the string is not a valid representation of a number.
 * @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.
 */
public actual fun String.toByte(radix: Int): Byte = toByteOrNull(radix) ?: numberFormatException(this)

package kotlin.text

/**
 * Parses the string as a signed [Short] number and returns the result.
 * @throws NumberFormatException if the string is not a valid representation of a number.
 */
public actual fun String.toShort(): Short = toShortOrNull() ?: numberFormatException(this)

package kotlin.text

/**
 * Parses the string as a signed [Byte] number and returns the result.
 * @throws NumberFormatException if the string is not a valid representation of a number.
 */
public actual fun String.toByteOrNull(): Byte

package kotlin.text

/**
 * Parses the string as a signed [Short] number and returns the result.
 * @throws NumberFormatException if the string is not a valid representation of a number.
 */
public actual fun String.toShortOrNull(): Short

package kotlin.text

/**
 * Parses the string as a signed [Byte] number and returns the result.
 * @throws NumberFormatException if the string is not a valid representation of a number.
 */
public actual fun String.toByteOrNull(radix: Int): Byte

package kotlin.text

package kotlin.text

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.text
IllegalArgumentException when [radix] is not a valid radix for string to number conversion.\n */
public actual fun
String.toShort(radix: Int): Short = toShortOrNull(radix) ?: numberFormatError(this)\n
/**
 * Parses the string as an [Int] number and returns the result.\n * @throws NumberFormatException if the string is not a valid representation of a number.\n * @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.\n */
public actual fun String.toInt(radix: Int): Int = toIntOrNull(radix) ?: numberFormatError(this)\n
/**
 * Parses the string as a [Long] number and returns the result.\n * @throws NumberFormatException if the string is not a valid representation of a number.\n * @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.\n */
public actual fun String.toLong(radix: Int): Long = toLongOrNull(radix) ?: numberFormatError(this)\n
/**
 * Parses the string as a [Double] number and returns the result.\n * @throws NumberFormatException if the string is not a valid representation of a number.\n * @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.\n */
public actual fun String.toDouble(): Double = (+(this.asDynamic())).unsafeCast<Double>().also {\n    if (it.isNaN() && !this.isNaN() \n        || it == 0.0 && this.isBlank())\n        numberFormatError(this)\n}

/**
 * Parses the string as a [Float] number and returns the result.\n * @throws NumberFormatException if the string is not a valid representation of a number.\n * @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.\n */
@kotlin.internal.InlineOnly
public actual inline fun String.toFloat(): Float =

toDouble().unsafeCast<Float>().also {\n    it = it / 0.0\n}

/**
 * Parses the string as a [Double] number and returns the result.\n * @throws NumberFormatException if the string is not a valid representation of a number.\n * @throws NumberFormatException if the string is not a valid representation of a number.\n */
public actual fun String.toDoubleOrNull(): Double? =

toDoubleOrNull().unsafeCast<Float>().takeIf {\n    !it.isNaN() \n}

/**
 * Returns a string representation of this [Byte] value in the specified [radix].\n * @throws IllegalArgumentException when [radix] is not a valid radix for number to string conversion.\n */
@SinceKotlin("1.2")
@kotlin.internal.InlineOnly
public actual inline fun Byte.toString(radix: Int): String = this.toInt().toString(radix)

/**
 * Returns a string representation of this [Short] value in the specified [radix].\n * @throws IllegalArgumentException when [radix] is not a valid radix for number to string conversion.\n */
@SinceKotlin("1.2")
@kotlin.internal.InlineOnly
public actual inline fun Short.toString(radix: Int): String = this.toInt().toString(radix)

/**
 * Returns a string representation of this [Int] value in the specified [radix].\n * @throws IllegalArgumentException when [radix] is not a valid radix for number to string conversion.\n */
@SinceKotlin("1.2")
public actual fun Int.toString(radix: Int): String =

@OptIn(ExperimentalStdlibApi::class)
private fun String.isNaN(): Boolean = when (this.lowercase()) {\n    "nan", "+nan", "-nan" -> true
    else -> false\n}

/**
 * Checks whether the given [radix] is valid radix for string to number and number to string conversion.\n */
private fun checkRadix(radix: Int): Int {
    if (radix !in 2..36) {
        throw IllegalArgumentException("radix $radix was not in valid range 2..36")
    }
    return radix
}

private fun digitOf(char: Char, radix: Int): Int = when {
    char >= '0' && char <= '9' -> char - '0'
    char >= 'A' && char <= 'Z' -> char - 'A' + 10
    char >= 'a' && char <= 'z' -> char - 'a' + 10
    else -> -1
}.let { if (it >= radix) -1 else it }

package kotlin.text

import kotlin.js.RegExp

/**
 * Provides enumeration values to use to set regular
 * expression options.
 */
public actual enum class RegexOption(val value: String) {
    /** Enables case-insensitive
     * matching. */
    IGNORE_CASE("i"),
    /** Enables multiline mode.
     * In multiline mode the
     * expressions `^` and `$`
     * match just after or just before,\n     * respectively, a line terminator or the end of the input
     * sequence. */
    MULTILINE("m")
}

/**
 * Represents the results from a single capturing group within a
 * [MatchResult] of [Regex].\n */
public actual data class

Open Source Used In security_automation identity-guest-cs 1626
**MatchGroup(actual val value: String)\n\n条例**

* Represents a compiled regular expression.\n
* Provides functions to match strings in text with a pattern, replace the found occurrences and split text around matches.\n
* For pattern syntax reference see [MDN RegExp](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/RegExp#Special_characters_meaning_in_regular_expressions)\n
* @constructor Creates a regular expression from the specified [pattern] string and the specified set of [options].\n
* /\n
* @constructor Regex(actual constructor(pattern: String, options: Set<RegexOption>)\n
* /\n
* /\n
* @constructor Regex(actual constructor(pattern: String, option: RegexOption) : this(pattern, setOf(option))\n
* /\n
* @constructor Regex(actual constructor(pattern: String) : this(pattern, emptySet())\n
* /\n
* The pattern string of this regular expression.\n
* /\n
* public actual val pattern: String = pattern\n
* /\n
* The set of options that were used to create this regular expression.\n
* /\n
* public actual val options: Set<RegexOption> = options.toSet()\n
* /\n
* private val nativePattern: RegExp = RegExp(pattern, options.map { it.value }.joinToString(separator = "+\") + "g")\n
* /\n
* Indicates whether the regular expression matches the entire [input].\n
* /\n
* private actual fun findInfix fun matches(input: CharSequence): Boolean {\n
*    nativePattern.reset()\n
*    val match = nativePattern.matchEntire(input.toString())\n
*    return match != null && match.index == 0 && match.index == 0 &\n
*    nativePattern.lastIndex == input.length\n
* /\n
* /\n
* Indicates whether the regular expression can find at least one match in the specified [input].\n
* /\n
* public actual fun containsMatchIn(input: CharSequence): Boolean {\n
*    nativePattern.reset()\n
*    return nativePattern.matchEntire(input.toString())\n
* /\n
* /\n
* Returns the first match of a regular expression in the [input], beginning at the specified [startIndex].\n
* /\n
* /\n
* @param startIndex An index to start search with, by default 0. Must not be less than zero and not greater than `input.length()`\n
* /\n
* @return An instance of [MatchResult] if match was found or null otherwise.\n
* /\n
* @throws IndexOutOfBoundsException if [startIndex] is less than zero or greater than the length of the [input] char sequence.\n
* /\n
* @Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\n
* /\n
* @actual fun find(input: CharSequence, startIndex: Int = 0): MatchResult? {\n
*    if (startIndex < 0 || startIndex > input.length) {\n
*        throw IndexOutOfBoundsException("Start index out of bounds: $startIndex, input length: ${input.length}")\n
*    }\n
*    val match = nativePattern.matchEntire(input.toString(), startIndex)\n
*    if (match != null) return match\n
* /\n
* /\n
* Returns a sequence of all occurrences of a regular expression within the [input] string, beginning at the specified [startIndex].\n
* /\n
* @Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\n
* /\n
* @actual fun findAll(input: CharSequence, startIndex: Int = 0): Sequence<MatchResult> {\n
*    if (startIndex < 0 || startIndex > input.length) {\n
*        throw IndexOutOfBoundsException("Start index out of bounds: $startIndex, input length: ${input.length}")\n
*    }\n
*    var match = find(input, startIndex)\n
*    if (match == null) return\n
* /\n
* /\n
* Attempts to match the entire [input] CharSequence against the pattern.\n
* /\n
* @param ret An instance of [MatchResult] if the entire input matches or null otherwise.\n
* /\n
* @param replacement A replacement expression that can include substitutions. See [String.prototype.replace](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/String/replace) function docs for details.\n
* /\n
* /\n
* replaces all occurrences of this regular expression in the specified [input] string with specified [replacement] expression.\n
* /\n
* /\n
* public actual fun replace(input: CharSequence, replacement: String): String = input.toString().nativeReplace(nativePattern, replacement)\n
* /\n
* /\n
* replaces all occurrences of this regular expression in the specified [input] string with the result of the given function [transform] that takes [MatchResult] and returns a string to be used as a replacement for that match.\n
* /\n
* public actual inline fun replace(input: CharSequence, transform: (MatchResult) -> CharSequence): String {\n
*    var match = find(input)\n
*    if (match == null) return
input.toString())

var lastStart = 0
val length = input.length
val sb = StringBuilder(length)

do {
    val foundMatch = match!!
    sb.append(input, lastStart, foundMatch.range.start)
    sb.append(transform(foundMatch))
    lastStart = foundMatch.range.endInclusive + 1
    match = foundMatch.next()
} while (lastStart < length && match != null)

if (lastStart < length) {
    sb.append(input, lastStart, length)
}

return sb.toString() /* Replaces the first occurrence of this regular expression in the specified [input] string with specified [replacement] expression. */

*param replacement A replacement expression that can include substitutions. See [String.prototype.replace](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/String/replace) function docs for details.

public actual fun replaceFirst(input: CharSequence, replacement: String): String {
    val nonGlobalOptions = options.map { it.value }.joinToString(separator = "")
    return input.toString().nativeReplace(RegExp(pattern, nonGlobalOptions), replacement) /* Splits the [input] CharSequence around matches of this regular expression. */

    *param limit Non-negative value specifying the maximum number of substrings the string can be split to.
    * Zero by default means no limit is set.

    @Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
    public actual fun split(input: CharSequence, limit: Int = 0): List<String> {
        require(limit >= 0) { "Limit must be non-negative, but was $limit" }
        val matches = findAll(input).let { if (limit == 0) it else it.take(limit - 1) }
        val result = mutableListOf<String>()

        var lastStart = 0

        for (match in matches) {
            result.add(input.subSequence(lastStart, match.range.start).toString())
            lastStart = match.range.endInclusive + 1
        }
        result.add(input.subSequence(lastStart, input.length).toString())
        return result
    }

    /**
    * Returns the string representation of this regular expression, namely the [pattern] of this regular expression.
    * Note that another regular expression constructed from the same pattern string may have different [options] and may match strings differently.

    public override fun toString(): String = nativePattern.toString()

    actual companion object {
        /**
        * Returns a regular expression that matches the specified [literal] string literally. No characters of that string will have special meaning when searching for an occurrence of the regular expression. *
        * public actual fun substitute(literal: String, replacement: String, caseInsensitive: Boolean): String = nonGlobalOptions.map { it.value }.joinToString(separator = "")
        */
        public actual fun fromLiteral(literal: String): Regex = Regex(escape(literal))

        /**
        * Returns a regular expression pattern string that matches the specified [literal] string literally. No characters of that string will have special meaning when searching for an occurrence of the regular expression. *
        * public actual fun substitute(literal: String, replacement: String, caseInsensitive: Boolean): String = nonGlobalOptions.map { it.value }.joinToString(separator = "")
        */
        public actual fun escape(literal: String): String = literal.nativeReplace(patternEscape, "\\$&")

        /**
        * Returns a literal replacement expression for the specified [literal] string. No characters of that string will have special meaning when it is used as a replacement string in [Regex.replace] function.
        */
        public actual fun escapeReplacement(literal: String): String = literal.nativeReplace(replacementEscape, "$$")

        private val patternEscape = RegExp("""\[^-\\^$*+?.()|\[\]{}\]""", "g")
        private val replacementEscape = RegExp("""\$""", "g")
    }
}

/**
* creates a regular expression from the specified [pattern] string and the specified single [option].
* level = DeprecationLevel.HIDDEN
* level = JsName("\"Regex_sd3q2\"\")
*/
@Deprecated("Provided for binary compatibility!", level = DeprecationLevel.HIDDEN)
@JsName("Regex_sd3q2\")
internal fun Regex_0(pattern: String, option: RegexOption): Regex = Regex(pattern, setOf(option))

/**
* creates a regular expression from the specified [pattern] string and the default options.
* level = DeprecationLevel.HIDDEN
* level = JsName("\"Regex_61poe\"\")
*/
@Deprecated("Provided for binary compatibility!", level = DeprecationLevel.HIDDEN)
@JsName("Regex_61poe\")
public fun Regex_1(pattern: String): Regex = Regex(pattern, emptySet())

/**
* creates a regular expression from the specified [pattern] string and the specified group values.
* level = DeprecationLevel.HIDDEN
* level = JsName("\"Regex_61poe\"\")
*/
@Deprecated("Provided for binary compatibility!", level = DeprecationLevel.HIDDEN)
@JsName("Regex_61poe\")
private fun Regex_2(pattern: String, emptySet(): Unit): Regex = Regex(pattern, emptySet())
override fun get(index: Int): String = match[index] ?: ""

override fun getFirst(): MatchResult? =
    this@findNext.findNext(input, if (range.isEmpty()) range.start + 1 else range.endInclusive + 1)

open class String

@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun String.matches(regex: String): Boolean {
    val result = this.match(regex)
    return result != null && result.size != 0
}

public fun String.matches(regex: String): Boolean {
    return when (regex.kind) {
        "[\w\d]+" ->
            KotlinStdlibJdk7RegexpImpl.regexMatches(input, regex, 0)
        else ->
            KotlinStdlibJdk7RegexpImpl.regexMatches(input, regex, 0)
    }
}

/**
 * Returns a copy of this string having its first letter titlecased using the rules of the default locale,
 * or the original string if it's empty or already starts with a title case letter.
 */
public actual fun String.capitalize(): String {
    return if (isNotEmpty()) substring(0, 1).toUpperCase() + substring(1) else this
}

/**
 * Returns a copy of this string having its first letter lowercased using the rules of the default locale,
 * or the original string if it's empty or already starts with a lower case letter.
 */
public actual fun String.decapitalize(): String {
    return if (isNotEmpty()) substring(0, 1).toLowerCase() + substring(1) else this
}

/**
 * Returns a copy of this string having its first letter lowercased using the rules of the default locale,
 * or the original string if it's empty or already starts with a lower case letter.
 */
public actual fun String.lowercase(): String {
    return if (isNotEmpty()) substring(0, 1).toLowerCase() + substring(1) else this
}

/**
 * Returns a copy of this string having its first letter uppercased using the rules of the default locale,
 * or the original string if it's empty or already starts with a title case letter.
 */
public actual fun String.capitalize(): String {
    return if (isNotEmpty()) substring(0, 1).toUpperCase() + substring(1) else this
}

/**
 * Returns a copy of this string having its first letter uppercased using the rules of the default locale,
 * or the original string if it's empty or already starts with a title case letter.
 */
public actual fun String.decapitalize(): String {
    return if (isNotEmpty()) substring(0, 1).toLowerCase() + substring(1) else this
}

/**
 * Returns a copy of this string having its first letter uppercased using the rules of the default locale,
 * or the original string if it's empty or already starts with a title case letter.
 */
public actual fun String.lowercase(): String {
    return if (isNotEmpty()) substring(0, 1).toLowerCase() + substring(1) else this
}

/**
 * Returns a copy of this string having its first letter uppercased using the rules of the default locale,
 * or the original string if it's empty or already starts with a title case letter.
 */
public actual fun String.capitalize(): String {
    return if (isNotEmpty()) substring(0, 1).toUpperCase() + substring(1) else this
}

/**
 * Returns a copy of this string having its first letter uppercased using the rules of the default locale,
 * or the original string if it's empty or already starts with a title case letter.
 */
public actual fun String.decapitalize(): String {
    return if (isNotEmpty()) substring(0, 1).toLowerCase() + substring(1) else this
}

/**
 * Returns a copy of this string having its first letter uppercased using the rules of the default locale,
 * or the original string if it's empty or already starts with a title case letter.
 */
public actual fun String.lowercase(): String {
    return if (isNotEmpty()) substring(0, 1).toLowerCase() + substring(1) else this
}

/**
 * Returns a copy of this string having its first letter uppercased using the rules of the default locale,
 * or the original string if it's empty or already starts with a title case letter.
 */
public actual fun String.capitalize(): String {
    return if (isNotEmpty()) substring(0, 1).toUpperCase() + substring(1) else this
}

/**
 * Returns a copy of this string having its first letter uppercased using the rules of the default locale,
 * or the original string if it's empty or already starts with a title case letter.
 */
public actual fun String.decapitalize(): String {
    return if (isNotEmpty()) substring(0, 1).toLowerCase() + substring(1) else this
}

/**
 * Returns a copy of this string having its first letter uppercased using the rules of the default locale,
 * or the original string if it's empty or already starts with a title case letter.
 */
public actual fun String.lowercase(): String {
    return if (isNotEmpty()) substring(0, 1).toLowerCase() + substring(1) else this
}

/**
 * Returns a copy of this string having its first letter uppercased using the rules of the default locale,
 * or the original string if it's empty or already starts with a title case letter.
 */
public actual fun String.capitalize(): String {
    return if (isNotEmpty()) substring(0, 1).toUpperCase() + substring(1) else this
}

/**
 * Returns a copy of this string having its first letter uppercased using the rules of the default locale,
 * or the original string if it's empty or already starts with a title case letter.
 */
public actual fun String.decapitalize(): String {
    return if (isNotEmpty()) substring(0, 1).toLowerCase() + substring(1) else this
}

/**
 * Returns a copy of this string having its first letter uppercased using the rules of the default locale,
 * or the original string if it's empty or already starts with a title case letter.
 */
public actual fun String.lowercase(): String {
    return if (isNotEmpty()) substring(0, 1).toLowerCase() + substring(1) else this
}

/**
 * Returns a copy of this string having its first letter uppercased using the rules of the default locale,
 * or the original string if it's empty or already starts with a title case letter.
 */
public actual fun String.capitalize(): String {
    return if (isNotEmpty()) substring(0, 1).toUpperCase() + substring(1) else this
}

/**
 * Returns a copy of this string having its first letter uppercased using the rules of the default locale,
 * or the original string if it's empty or already starts with a title case letter.
 */
public actual fun String.decapitalize(): String {
    return if (isNotEmpty()) substring(0, 1).toLowerCase() + substring(1) else this
}

/**
 * Returns a copy of this string having its first letter uppercased using the rules of the default locale,
 * or the original string if it's empty or already starts with a title case letter.
 */
public actual fun String.lowercase(): String {
    return if (isNotEmpty()) substring(0, 1).toLowerCase() + substring(1) else this
}

/**
 * Returns a copy of this string having its first letter uppercased using the rules of the default locale,
 * or the original string if it's empty or already starts with a title case letter.
 */
public actual fun String.capitalize(): String {
    return if (isNotEmpty()) substring(0, 1).toUpperCase() + substring(1) else this
}

/**
 * Returns a copy of this string having its first letter uppercased using the rules of the default locale,
 * or the original string if it's empty or already starts with a title case letter.
 */
public actual fun String.decapitalize(): String {
    return if (isNotEmpty()) substring(0, 1).toLowerCase() + substring(1) else this
}

/**
 * Returns a copy of this string having its first letter uppercased using the rules of the default locale,
 * or the original string if it's empty or already starts with a title case letter.
 */
public actual fun String.lowercase(): String {
    return if (isNotEmpty()) substring(0, 1).toLowerCase() + substring(1) else this
}

/**
 * Returns a copy of this string having its first letter uppercased using the rules of the default locale,
 * or the original string if it's empty or already starts with a title case letter.
 */
public actual fun String.capitalize(): String {
    return if (isNotEmpty()) substring(0, 1).toUpperCase() + substring(1) else this
}

/**
 * Returns a copy of this string having its first letter uppercased using the rules of the default locale,
 * or the original string if it's empty or already starts with a title case letter.
 */
public actual fun String.decapitalize(): String {
    return if (isNotEmpty()) substring(0, 1).toLowerCase() + substring(1) else this
}

/**
 * Returns a copy of this string having its first letter uppercased using the rules of the default locale,
 * or the original string if it's empty or already starts with a title case letter.
 */
public actual fun String.lowercase(): String {
    return if (isNotEmpty()) substring(0, 1).toLowerCase() + substring(1) else this
}

/**
 * Returns a copy of this string having its first letter uppercased using the rules of the default locale,
 * or the original string if it's empty or already starts with a title case letter.
 */
public actual fun String.capitalize(): String {
    return if (isNotEmpty()) substring(0, 1).toUpperCase() + substring(1) else this
}

/**
 * Returns a copy of this string having its first letter uppercased using the rules of the default locale,
 * or the original string if it's empty or already starts with a title case letter.
 */
public actual fun String.decapitalize(): String {
    return if (isNotEmpty()) substring(0, 1).toLowerCase() + substring(1) else this
}

/**
 * Returns a copy of this string having its first letter uppercased using the rules of the default locale,
 * or the original string if it's empty or already starts with a title case letter.
 */
public actual fun String.lowercase(): String {
    return if (isNotEmpty()) substring(0, 1).toLowerCase() + substring(1) else this
}

/**
 * Returns a copy of this string having its first letter uppercased using the rules of the default locale,
 * or the original string if it's empty or already starts with a title case letter.
 */
public actual fun String.capitalize(): String {
    return if (isNotEmpty()) substring(0, 1).toUpperCase() + substring(1) else this
}

/**
 * Returns a copy of this string having its first letter uppercased using the rules of the default locale,
 * or the original string if it's empty or already starts with a title case letter.
 */
public actual fun String.decapitalize(): String {
    return if (isNotEmpty()) substring(0, 1).toLowerCase() + substring(1) else this
}

/**
 * Returns a copy of this string having its first letter uppercased using the rules of the default locale,
 * or the original string if it's empty or already starts with a title case letter.
 */
public actual fun String.lowercase(): String {
    return if (isNotEmpty()) substring(0, 1).toLowerCase() + substring(1) else this
}

/**
 * Returns a copy of this string having its first letter uppercased using the rules of the default locale,
 * or the original string if it's empty or already starts with a title case letter.
 */
public actual fun String.capitalize(): String {
    return if (isNotEmpty()) substring(0, 1).toUpperCase() + substring(1) else this
}

/**
 * Returns a copy of this string having its first letter uppercased using the rules of the default locale,
 * or the original string if it's empty or already starts with a title case letter.
 */
public actual fun String.decapitalize(): String {
    return if (isNotEmpty()) substring(0, 1).toLowerCase() + substring(1) else this
}

/**
 * Returns a copy of this string having its first letter uppercased using the rules of the default locale,
 * or the original string if it's empty or already starts with a title case letter.
 */
public actual fun String.lowercase(): String {
    return if (isNotEmpty()) substring(0, 1).toLowerCase() + substring(1) else this
}
nativeReplacer(RegExp(Regex.escape(oldValue), if (ignoreCase) "gi" else "g"),
    newChar.toString()),@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
    public actual fun String.replace(oldChar: Char, newChar: Char, ignoreCase: Boolean = false): String =
    nativeReplacer(RegExp(Regex.escape(oldChar.toString()), if (ignoreCase) "gi" else "g"),
    newChar.toString()),@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
    public actual fun String.replaceFirst(oldValue: String, newValue: String, ignoreCase: Boolean = false): String =
    nativeReplacer(RegExp(Regex.escape(oldValue), if (ignoreCase) "i" else ""),
    newValue),/* Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language
    contributors
    Use of this source code is governed by the Apache 2.0 license that can be found in the
    license/LICENSE.txt file. */
    package kotlin.text
    /** Returns the negative [size] if [throwOnMalformed] is
     false, throws [CharacterCodingException] otherwise. */
    private fun malformed(size: Int, index: Int, throwOnMalformed: Boolean): Int {
        if (throwOnMalformed) throw
            CharacterCodingException("Malformed sequence starting at ${index - 1}")
        return -size
    }
    /** Returns code point corresponding to UTF-16 surrogate pair, where the first of the pair is the [high] and the second is in
     the [string] at the [index]. Returns zero if the pair is malformed and [throwOnMalformed] is false. */
    private fun codePointFromSurrogate(string: String, high: Int, index: Int, endIndex: Int, throwOnMalformed: Boolean): Int {
        if (high !in 0xD800..0xDBFF || index >= endIndex) {
            return malformed(0, index, throwOnMalformed)
        }
        val low = string[index].toInt()
        if (low !in 0xDC00..0xDFFF) {
            return malformed(0, index, throwOnMalformed)
        }
        return 0x10000 + ((high and 0x3FF) shl 10) or (low and 0x3FF)
    }
    /** Returns code point corresponding to UTF-8 sequence of two bytes, where the first byte of the sequence is the [byte1] and the second byte is in [bytes] array at the [index]. Returns zero if the sequence is malformed and [throwOnMalformed] is false. */
    private fun codePointFrom2(bytes: ByteArray, byte1: Int, index: Int, endIndex: Int, throwOnMalformed: Boolean): Int {
        if (byte1 and 0x1E == 0 || index >= endIndex) {
            return malformed(0, index, throwOnMalformed)
        }
        val byte2 = bytes[index].toInt()
        if (byte2 and 0xC0 != 0x80) {
            return malformed(0, index, throwOnMalformed)
        }
        return (byte1 shl 6) xor byte2 xor -0x1E080
    }
    /** Returns code point corresponding to UTF-8 sequence of three bytes, where the first byte of the sequence is the [byte1] and the others are in [bytes] array starting from the [index]. Returns a non-positive value indicating number of bytes from [bytes] included in malformed sequence if the sequence is malformed and [throwOnMalformed] is false. */
    private fun codePointFrom3(bytes: ByteArray, byte1: Int, index: Int, endIndex: Int, throwOnMalformed: Boolean): Int {
        if (index >= endIndex) {
            return malformed(0, index, throwOnMalformed)
        }
        val byte2 = bytes[index].toInt()
        if (byte1 and 0xF == 0) {
            if (byte2 and 0xEF == 0xA0) {
                // Non-shortest form
                return malformed(0, index, throwOnMalformed)
            } else if (byte1 and 0x0F == 0xD) {
                return (byte1 shl 6) xor byte2 xor 0xF80
            } else if (byte1 and 0xF == 0xD) {
                return (byte1 shl 12) xor byte2 xor -0x1E080
            } else if (byte1 and 0xF == 0x0) {
                return (byte1 shl 18) xor byte2 xor -0xEF080
            }
        } else if (byte2 and 0x0F == 0x0) {
            return (byte2 and 0xEF0) xor 0x1E080
        } else if (byte1 and 0xEF == 0xA0) {
            return (byte1 and 0x0F0) xor 0xEF0
        } else if (byte1 and 0x0F == 0xD) {
            return (byte1 and 0xF0) xor 0x1E0
        } else if (byte1 and 0xF == 0x0) {
            return (byte1 and 0x0F) xor 0xF
        }
        return (byte1 shl 12) xor byte2 xor -0x1E080
    }
    /** Returns code point corresponding to UTF-8 sequence of four bytes, where the first byte of the sequence is the [byte1] and the others are in [bytes] array starting from the [index]. Returns a non-positive value indicating number of bytes from [bytes] included in malformed sequence if the sequence is malformed and [throwOnMalformed] is false. */
        if (index >= endIndex) {
            return malformed(0, index, throwOnMalformed)
        }
        val byte2 = bytes[index].toInt()
        if (byte1 and 0xF == 0) {
            if (byte2 and 0xEF == 0xA0) {
                return (byte1 and 0x0F0) xor 0xEF0
            }
        } else if (byte1 and 0xF == 0xD) {
            return (byte1 and 0xF0) xor 0x1E0
        } else if (byte1 and 0xF == 0x0) {
            return (byte1 and 0x0F) xor 0xF
        }
        return (byte1 and 0x0EF) xor byte2 xor -0x1E080
ByteArray, byte1: Int, index: Int, throwOnMalformed: Boolean): Int {
    if (index >= endIndex) {
        throwOnMalformed(0, index, throwOnMalformed)
    } else if (byte1 and 0xF == 0x0) {
        return malformed(0, index, throwOnMalformed)
    } else if (byte1 and 0xF > 0x4) {
        return malformed(0, index, throwOnMalformed)
    } else if (index + 1 == endIndex) {
        return malformed(1, index, throwOnMalformed)
    } else if (index + 2 == endIndex) {
        return malformed(2, index, throwOnMalformed)
    } else if (byte2 and 0xC0 != 0x80) {
        return malformed(2, index, throwOnMalformed)
    } else if (byte3 and 0xC0 != 0x80) {
        return malformed(3, index, throwOnMalformed)
    } else if (byte4 and 0xC0 != 0x80) {
        return malformed(4, index, throwOnMalformed)
    } else {
        // Non-shortest form
        return malformed(0, index, throwOnMalformed)
    }
}

/**
 * Maximum number of bytes needed to encode a single char.
 * Code points in `0..0x7F` are encoded in a single byte.
 * Code points in `0x80..0x7FF` are encoded in two bytes.
 * Code points in `0x800..0xD7FF` or in `0xE000..0xFFFF` are encoded in three bytes.
 * Surrogate code points in `0xD800..0xDFFF` are not Unicode scalar values, therefore aren't encoded.
 * Code points in `0x10000..0x10FFFF` are represented by a pair of surrogate 'Char's and are encoded in four bytes.
 */
private const val MAX_BYTES_PER_CHAR = 3

private val REPLACEMENT_BYTE_SEQUENCE: ByteArray = byteArrayOf(0xEF.toByte(), 0xBF.toByte(), 0xBD.toByte())

/**
 * Encodes the [string] using UTF-8 and returns the resulting [ByteArray].
 * @param string the string to encode.
 * @param startIndex the start offset (inclusive) of the substring to encode.
 * @param endIndex the end offset (exclusive) of the substring to encode.
 * @param throwOnMalformed whether to throw on malformed char sequence or replace by the [REPLACEMENT_BYTE_SEQUENCE].
 * @throws CharacterCodingException if the char sequence is malformed and [throwOnMalformed] is true.
 */
internal fun encodeUtf8(string: String, startIndex: Int, endIndex: Int, throwOnMalformed: Boolean): ByteArray {
    require(startIndex >= 0 && endIndex <= string.length && startIndex <= endIndex)

    val bytes = ByteArray((endIndex - startIndex) * MAX_BYTES_PER_CHAR)
    var byteIndex = 0
    var charIndex = startIndex

    while (charIndex < endIndex) {
        val code = string[charIndex++].toInt()
        when {
            code < 0x80 ->
                bytes[byteIndex++] = code.toByte()
            code < 0x800 -> {
                bytes[byteIndex++] = ((code shr 6) or 0xC0).toByte()
                bytes[byteIndex++] = ((code and 0x3F) or 0x80).toByte()
            }
            code < 0xD800 || code >= 0xE000 -> {
                bytes[byteIndex++] = ((code shr 12) or 0xE0).toByte()
                bytes[byteIndex++] = (((code shr 6) and 0x3F) or 0x80).toByte()
                bytes[byteIndex++] = ((code and 0x3F) or 0x80).toByte()
            }
            else -> { // Surrogate char value
                val codePoint = codePointFromSurrogate(string, code, charIndex, startIndex, endIndex, throwOnMalformed)
                if (codePoint <= 0) {
                    bytes[byteIndex++] = REPLACEMENT_CHAR
                } else {
                    bytes[byteIndex++] = ((codePoint shr 18) or 0x80).toByte()
                    bytes[byteIndex++] = (((codePoint shr 12) and 0x3F) or 0x80).toByte()
                    bytes[byteIndex++] = ((codePoint shr 6) and 0x3F) or 0x80).toByte()
                    bytes[byteIndex++] = (codePoint and 0x3F) or 0x80).toByte()
                    charIndex++
                    charIndex++
                    charIndex++
                    byteIndex++
                    byteIndex++
                    byteIndex++
                    byteIndex++
                }
            }
        }
    }

    return if (bytes.size == byteIndex) bytes else bytes.copyOf(byteIndex)
}

/**
 * The character a malformed UTF-8 byte sequence is replaced by.
 */
private const val REPLACEMENT_CHAR = '\uFFFD'

/**
 * Decodes the UTF-8 [bytes] array and returns the resulting [String].
 * @param bytes the byte array to decode.
 * @param throwOnMalformed whether to throw on malformed byte sequence or replace by the [REPLACEMENT_CHAR].
 * @throws CharacterCodingException if the array is malformed UTF-8 byte sequence and [throwOnMalformed] is true.
 */
internal fun decodeUtf8(bytes: ByteArray,
if (indent.isNotEmpty()) {
    val messageLines = if (stackStart == 0) 0 else 1 + shortInfo.count { c -> c == '\n' }
    stack.lineSequence().forEachIndexed { index, line ->
        if (index >= messageLines)
            target.append(indent)
        target.append(line).append("\n")
    }
}

val suppressed = suppressedExceptions
if (suppressed.isNotEmpty()) {
    val suppressedIndent = indent + "    
    \n    \n    for (s in suppressed) {
        s.dumpFullTrace(suppressedIndent,"Suppressed: ")
    }
}
return true
}

private fun dropCommonFrames(stack: String,
    stackStart: Int): String {
    var commonFrames: Int = 0
    var lastBreak: Int = 0
    var preLastBreak: Int = 0
    for (pos in 0 until minOf(topStack.length - topStackStart, stack.length - stackStart)) {
        val c = stack[stack.lastIndex - pos]
        if (c != topStack[topStack.lastIndex - pos]) break
        if (c == '\n') {
            commonFrames += 1
            preLastBreak = lastBreak
            lastBreak = pos
        }
    }
    if (commonFrames <= 1) return stack
    while (preLastBreak > 0 && stack[stack.lastIndex - (preLastBreak - 1)] == ' ')
        preLastBreak -= 1
    return stack.dropLast(preLastBreak) + "... and ${commonFrames - 1} more common stack frames skipped"
}

"/*
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.time

@SinceKotlin("1.3")
@ExperimentalTime
public actual enum class DurationUnit(internal val scale: Double) {
    /**
     * Time unit representing one nanosecond, which is 1/1000 of
     * a microsecond.
     */
    NANOSECONDS(1e0),
    /**
     * Time unit representing one microsecond, which is
     * 1/1000 of a millisecond.
     */
    MICROSECONDS(1e3),
    /**
     * Time unit representing one millisecond,
     */
    MILLISECONDS(1e6),
    /**
     * Time unit representing one second.
     */
    SECONDS(1e9),
    /**
     * Time unit representing one minute.
     */
    MINUTES(60e9),
    /**
     * Time unit representing one hour.
     */
    HOURS(3600e9),
    /**
     * Time unit representing one day,
     */
    DAYS(86400e9);
}

@SinceKotlin("1.3")
@ExperimentalTime
internal actual fun convertDurationUnit(value: Double, sourceUnit: DurationUnit, targetUnit: DurationUnit): Double {
    val sourceCompareTarget = sourceUnit.scale.compareTo(targetUnit.scale)
    return when {
        sourceCompareTarget > 0 -> value * (sourceUnit.scale / targetUnit.scale)
        sourceCompareTarget < 0 -> value /
            (targetUnit.scale / sourceUnit.scale)
        else -> value
    }
}

"/*
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.time

import org.w3c.performance.GlobalPerformance
import org.w3c.performance.Performance

@SinceKotlin("1.3")
@ExperimentalTime
internal actual object MonotonicTimeSource : TimeSource {

    private val actualSource: TimeSource = run {
        val isNode:
            Boolean = js("typeof process !== 'undefined' && process.versions && !!process.versions.node")
        if
            isNode
            HrTimeSource(js("process").unsafeCast<Process>())
        else
            js("self").unsafeCast<GlobalPerformance?>()?.performance?.let(::PerformanceTimeSource)
                ?:
            DateNowTimeSource

    }

    override fun markNow(): TimeMark = actualSource.markNow()
            return when when

    }

    override fun read(): Double = actualSource.read()
    override fun toString(): String = actualSource.toString()
}
object DateNowTimeSource : AbstractDoubleTimeSource(unit = DurationUnit.MILLISECONDS) {
    override fun read(): Double = kotlin.js.Date.now()
    override fun toString(): String = "TimeSource(Date.now())"
}

/*
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.time

import kotlin.js.json
import kotlin.math.*

internal actual fun formatToExactDecimals(value: Double, decimals: Int): String {
    val rounded = if (decimals == 0) {
        value
    } else {
        val pow = 10.0.pow(decimals)
        kotlin.js.Math.round(abs(value) * pow) / pow * sign(value)
    }
    return rounded.asDynamic().toFixed(decimals).unsafeCast<String>()
}

internal actual fun formatUpToDecimals(value: Double, decimals: Int): String {
    return value.asDynamic().toLocaleString("en-us", json("maximumFractionDigits" to decimals)).unsafeCast<String>()
}

internal actual fun formatScientific(value: Double): String {
    return value.asDynamic().toExponential(2).unsafeCast<String>()
}

/*
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlinx.dom

import org.w3c.dom.*
import kotlin.contracts.*

/**
 * Creates a new element with the specified [name].
 * The element is initialized with the specified [init] function.
 * @param name the element name
 * @param init the element initializer
 * @return the newly created element
 */
@SinceKotlin("1.4")
public fun Document.createElement(name: String, init: Element.() -> Unit): Element {
    contract { callsInPlace(init, InvocationKind.EXACTLY_ONCE) }
    return createElement(name).apply(init)
}

/**
 * Appends a newly created element with the specified [name] to this element.
 * The element is initialized with the specified [init] function.
 * @param name the element name
 * @param init the element initializer
 * @return the newly created element
 */
@SinceKotlin("1.4")
public fun Element.appendElement(name: String, init: Element.() -> Unit): Element {
    contract { callsInPlace(init, InvocationKind.EXACTLY_ONCE) }
    return ownerDocument!!.createElement(name, init).also {
        appendChild(it)
    }
}

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlinx.text

/**
 * Converts the string into a regular expression [Regex] with the default options.
 */
@kotlin.internal.InlineOnly
public inline fun String.toRegex(): Regex = Regex(this)

/**
 * Converts the string into a regular expression [Regex] with the specified single [option].
 */
@kotlin.internal.InlineOnly
public inline fun String.toRegex(option: RegexOption): Regex = Regex(this, option)

/**
 * Converts the string into a regular expression [Regex] with the specified set of [options].
 */
@kotlin.internal.InlineOnly
public inline fun String.toRegex(options: Set<RegexOption>): Regex = Regex(this, options)

/*
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */
starting from the specified [startIndex] toward the beginning.

* `Returns `-1` if the specified [string] does not occur in this string builder starting at the specified [startIndex].
* `@SinceKotlin("1.4")`
* `@WasExperimental(ExperimentalStdlibApi::class)`

// **
* Inserts the string representation of the specified boolean [value] into this string builder at the specified [index] and returns this instance.
* `@throws IndexOutOfBoundsException or [IllegalArgumentException] if [index] is less than zero or greater than the length of this string builder."
* `@SinceKotlin("1.4")`
* `@WasExperimental(ExperimentalStdlibApi::class)`
* `fun insert(index: Int, value: Boolean): StringBuilder`
* `/**
* Inserts characters in the specified character array [value] into this string builder at the specified [index] and returns this instance."
* `@throws IndexOutOfBoundsException if [index] is less than zero or greater than the length of this string builder."
* `@SinceKotlin("1.4")`
* `@WasExperimental(ExperimentalStdlibApi::class)`
* `fun insert(index: Int, value: CharArray): StringBuilder`
* `/**
* Inserts characters in the specified character array [value] into this string builder at the specified [index] and returns this instance."
* `@throws IndexOutOfBoundsException if [index] is less than zero or greater than the length of this string builder."
* `@SinceKotlin("1.4")`
* `@WasExperimental(ExperimentalStdlibApi::class)`
* `fun insert(index: Int, value: Char): StringBuilder`
* `/**
* Inserts the string representation of the specified boolean [value] into this string builder at the specified [index] and returns this instance."
* `@throws IndexOutOfBoundsException or [IllegalArgumentException] if [index] is less than zero or greater than the length of this string builder."
* `@SinceKotlin("1.4")`
* `@WasExperimental(ExperimentalStdlibApi::class)`
* `fun insert(index: Int, value: Any?): StringBuilder`
* `/**
* Inserts the string representation of the specified character sequence [value] into this string builder at the specified [index] and returns this instance."
* `@throws IndexOutOfBoundsException or [IllegalArgumentException] if [index] is less than zero or greater than the length of this string builder."
* `@SinceKotlin("1.4")`
* `@WasExperimental(ExperimentalStdlibApi::class)`
* `fun insert(index: Int, value: String?): StringBuilder`
* `/**
* Inserts the string representation of the specified character sequence [value] into this string builder at the specified [index] and returns this instance."
* `@throws IndexOutOfBoundsException or [IllegalArgumentException] if [index] is less than zero or greater than the length of this string builder."
* `@SinceKotlin("1.4")`
* `@WasExperimental(ExperimentalStdlibApi::class)`
* `fun insert(index: Int, value: CharSequence?): StringBuilder`
* `/**
* Inserts characters in the specified character array [value] into this string builder at the specified [index] and returns this instance."
* `@SinceKotlin("1.4")`
* `@WasExperimental(ExperimentalStdlibApi::class)`
* `fun insert(index: Int, value: String): StringBuilder`
* `/**
* Inserts characters in the specified character array [value] into this string builder at the specified [index] and returns this instance."
* `@SinceKotlin("1.4")`
* `@WasExperimental(ExperimentalStdlibApi::class)`
* `fun insert(index: Int, value: Int): StringBuilder`
* `/**
* Inserts a single character at the specified [index] and returns this instance."
* `@Throws IndexOutOfBoundsException if [index] is less than zero or greater than the length of this string builder."
* `@SinceKotlin("1.4")`
* `@WasExperimental(ExperimentalStdlibApi::class)`
* `fun insert(index: Int, value: Boolean): StringBuilder`
* `/**
* Inserts a single boolean at the specified [index] and returns this instance."
* `@SinceKotlin("1.4")`
* `@WasExperimental(ExperimentalStdlibApi::class)`
* `fun insert(index: Int, value: Char): StringBuilder`
* `/**
* Inserts a single character at the specified [index] and returns this instance."
* `@SinceKotlin("1.4")`
* `@WasExperimental(ExperimentalStdlibApi::class)`
* `fun insert(index: Int, value: CharArray): StringBuilder`
* `/**
* Inserts characters in the specified character array [value] into this string builder at the specified [index] and returns this instance."
* `@Throws IndexOutOfBoundsException if [index] is less than zero or greater than the length of this string builder."
* `@SinceKotlin("1.4")`
* `@WasExperimental(ExperimentalStdlibApi::class)`
* `fun insert(index: Int, value: Any?): StringBuilder`
* `/**
* Inserts characters in the specified character array [value] into this string builder at the specified [index] and returns this instance."
* `@Throws IndexOutOfBoundsException if [index] is less than zero or greater than the length of this string builder."
* `@SinceKotlin("1.4")`
* `@WasExperimental(ExperimentalStdlibApi::class)`
* `fun insert(index: Int, value: String?): StringBuilder`
* `/**
* Inserts characters in the specified character array [value] into this string builder at the specified [index] and returns this instance."
* `@Throws IndexOutOfBoundsException if [index] is less than zero or greater than the length of this string builder."
* `@SinceKotlin("1.4")`
* `@WasExperimental(ExperimentalStdlibApi::class)`
* `fun insert(index: Int, value: CharSequence?): StringBuilder`
* `/**
* Inserts characters in the specified character array [value] into this string builder at the specified [index] and returns this instance."
* `@Throws IndexOutOfBoundsException if [index] is less than zero or greater than the length of this string builder."
* `@SinceKotlin("1.4")`
* `@WasExperimental(ExperimentalStdlibApi::class)`
* `fun insert(index: Int, value: Int): StringBuilder`
* `/**
* Inserts a single character at the specified [index] and returns this instance."
* `@SinceKotlin("1.4")`
* `@WasExperimental(ExperimentalStdlibApi::class)`
* `fun insert(index: Int, value: Boolean): StringBuilder`
* `/**
* Inserts a single boolean at the specified [index] and returns this instance."
* `@SinceKotlin("1.4")`
* `@WasExperimental(ExperimentalStdlibApi::class)`
* `fun insert(index: Int, value: Char): StringBuilder`
* `/**
* Inserts a single character at the specified [index] and returns this instance."
* `@SinceKotlin("1.4")`
* `@WasExperimental(ExperimentalStdlibApi::class)`
* `fun insert(index: Int, value: CharArray): StringBuilder`
* `/**
* Inserts characters in the specified character array [value] into this string builder at the specified [index] and returns this instance."
* `@SinceKotlin("1.4")`
* `@WasExperimental(ExperimentalStdlibApi::class)`
* `fun insert(index: Int, value: Any?): StringBuilder`
* `/**
* Inserts characters in the specified character array [value] into this string builder at the specified [index] and returns this instance."
* `@SinceKotlin("1.4")`
* `@WasExperimental(ExperimentalStdlibApi::class)`
* `fun insert(index: Int, value: String?): StringBuilder`
* `/**
* Inserts characters in the specified character array [value] into this string builder at the specified [index] and returns this instance."
* `@SinceKotlin("1.4")`
* `@WasExperimental(ExperimentalStdlibApi::class)`
* `fun insert(index: Int, value: CharSequence?): StringBuilder`
* `/**
* Inserts characters in the specified character array [value] into this string builder at the specified [index] and returns this instance."
* `@SinceKotlin("1.4")`
* `@WasExperimental(ExperimentalStdlibApi::class)`
* `fun insert(index: Int, value: Int): StringBuilder`
* `/**
* Inserts a single character at the specified [index] and returns this instance."
* `@SinceKotlin("1.4")`
* `@WasExperimental(ExperimentalStdlibApi::class)`
* `fun insert(index: Int, value: Boolean): StringBuilder`
* `/**
* Inserts a single boolean at the specified [index] and returns this instance."
* `@SinceKotlin("1.4")`
* `@WasExperimental(ExperimentalStdlibApi::class)`
* `fun insert(index: Int, value: Char): StringBuilder`
* `/**
* Inserts a single character at the specified [index] and returns this instance."
* `@SinceKotlin("1.4")`
* `@WasExperimental(ExperimentalStdlibApi::class)`
* `fun insert(index: Int, value: CharArray): StringBuilder`
* `/**
* Inserts characters in the specified character array [value] into this string builder at the specified [index] and returns this instance."
* `@SinceKotlin("1.4")`
* `@WasExperimental(ExperimentalStdlibApi::class)`
* `fun insert(index: Int, value: Char): StringBuilder`
*un * @throws IndexOutOfBoundsException or [IllegalArgumentOutOfRangeException] when [startIndex] or [endIndex] is out of range of the [value] character sequence indices or when `startIndex > endIndex`.

*un * @param index the position in this string builder to insert at.
*un * @param value the array from which characters are inserted.
*un * @param startIndex the beginning (inclusive) of the subarray to insert.
*un * @param endIndex the end (exclusive) of the subarray to insert.
*un * @throws IndexOutOfBoundsException or [IllegalArgumentOutOfRangeException] when [startIndex] or [endIndex] is out of range of the [value] array indices or when `startIndex > endIndex`.

*un * @param index the position in this string builder at.
*un * @param value the character sequence from which a subsequence is inserted.
*un * @param startIndex the beginning (inclusive) of the subsequence to insert.
*un * @param endIndex the end (exclusive) of the subsequence to insert.
*un * @throws IndexOutOfBoundsException or [IllegalArgumentOutOfRangeException] when [startIndex] or [endIndex] is out of range of the [value] character sequence indices or when `startIndex > endIndex`.

*un * @param index the position in this string builder to insert at.
*un * @param value the array from which characters are inserted.
*un * @param startIndex the beginning (inclusive) of the subarray to insert.
*un * @param endIndex the end (exclusive) of the subarray to insert.
*un * @throws IndexOutOfBoundsException or [IllegalArgumentOutOfRangeException] when [startIndex] or [endIndex] is out of range of the [value] array indices or when `startIndex > endIndex`.

*/
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public expect fun 
StringBuilder.appendRange(value: CharSequence, startIndex: Int, endIndex: Int): StringBuilder

/**
 * Inserts characters in a subarray of the specified character array [value] into this string builder at the specified [index] and returns this instance.
*un * The inserted characters go in same order as in the [value] array, starting at [index].
*un * @param index the position in this string builder to insert at.
*un * @param value the array from which characters are inserted.
*un * @param startIndex the beginning (inclusive) of the subarray to insert.
*un * @param endIndex the end (exclusive) of the subarray to insert.
*un * @throws IndexOutOfBoundsException or [IllegalArgumentOutOfRangeException] when [startIndex] or [endIndex] is out of range of the [value] array indices or when `startIndex > endIndex`.

*/
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public expect fun 
StringBuilder.insertRange(index: Int, value: CharArray, startIndex: Int, endIndex: Int): StringBuilder

/**
 * Inserts characters in a subsequence of the specified character sequence [value] into this string builder at the specified [index] and returns this instance.
*un * The inserted characters go in same order as in the [value] character sequence, starting at [index].
*un * @param index the position in this string builder to insert at.
*un * @param value the character sequence from which a subsequence is inserted.
*un * @param startIndex the beginning (inclusive) of the subsequence to insert.
*un * @param endIndex the end (exclusive) of the subsequence to insert.
*un * @throws IndexOutOfBoundsException or [IllegalArgumentOutOfRangeException] when [startIndex] or [endIndex] is out of range of the [value] character sequence indices or when `startIndex > endIndex`.

*/
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public expect fun 
StringBuilder.insertRange(index: Int, value: CharSequence, startIndex: Int, endIndex: Int): StringBuilder

@Suppress("EXTENSION_SHADOWED_BY_MEMBER")
@Deprecated("Use append(value: Any?) instead", ReplaceWith("append(value = obj)"), DeprecationLevel.WARNING)
@kotlin.internal.InlineOnly
public inline fun StringBuilder.append(obj: Any?): StringBuilder = this.append(obj)

/**
 * Builds new string by populating newly created [StringBuilder] using provided [builderAction] and then converting it to [String].
 */
@kotlin.internal.InlineOnly
public inline fun 
StringBuilder.buildString(builderAction: StringBuilder.() -> Unit): String {
    contract { callsInPlace(builderAction, InvocationKind.EXACTLY_ONCE) }
    return StringBuilder().apply(builderAction).toString()
}

/**
 * Builds new string by populating newly created [StringBuilder] initialized with the given [capacity] using provided [builderAction] and then converting it to [String].
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public inline fun buildString(capacity: Int, builderAction: StringBuilder.() -> Unit): String {
    contract { callsInPlace(builderAction, InvocationKind.EXACTLY_ONCE) }
    return StringBuilder(capacity).apply(builderAction).toString()
}

/** Appends all arguments to the given [StringBuilder].
 */
public fun StringBuilder.append(vararg value: String?): StringBuilder {
    for (item in value)
        append(item)
    return this
}

/** Appends all arguments to the given [StringBuilder].
 */
public fun StringBuilder.append(vararg value: Any?): StringBuilder {
    for (item in value)
        append(item)
    return this
}

/** Appends a line feed character (`\n`) to this [StringBuilder].
 */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun StringBuilder.appendLine(): 
StringBuilder = append('\n')

/** Appends [value] to this [StringBuilder], followed by a line feed character (`\n`).
 */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun StringBuilder.appendLine(value: CharSequence?): 
StringBuilder = append(value).appendLine()
preferLowPowerToHighPerformance: Boolean? = false, failIfMajorPerformanceCaveat: Boolean? = false):

WebGLContextAttributes {'val o = js("{"[]")'\n  o['\"alpha\"] = alpha'\n  o['\"depth\"] = depth'\n  o['\"stencil\"] = stencil'\n  o['\"antialias\"] = antialias'\n  o['\"premultipliedAlpha\"] = premultipliedAlpha'\n  o['\"preserveDrawingBuffer\"] = preserveDrawingBuffer'\n  o['\"preferLowPowerToHighPerformance\"] = preferLowPowerToHighPerformance'\n  o['\"failIfMajorPerformanceCaveat\"] = failIfMajorPerformanceCaveat'\n
return o'

public external abstract class WebGLObject

/*
 * Exposes the JavaScript

[WebGLBuffer](https://developer.mozilla.org/en/docs/Web/API/WebGLBuffer) to Kotlin
*/
public external abstract class WebGLBuffer : WebGLObject

/*
 * Exposes the JavaScript

[WebGLFramebuffer](https://developer.mozilla.org/en/docs/Web/API/WebGLFramebuffer) to Kotlin
*/
public external abstract class WebGLFramebuffer : WebGLObject

/*
 * Exposes the JavaScript

[WebGLProgram](https://developer.mozilla.org/en/docs/Web/API/WebGLProgram) to Kotlin
*/
public external abstract class WebGLProgram : WebGLObject

/*
 * Exposes the JavaScript

[WebGLRenderbuffer](https://developer.mozilla.org/en/docs/Web/API/WebGLRenderbuffer) to Kotlin
*/
public external abstract class WebGLRenderbuffer : WebGLObject

/*
 * Exposes the JavaScript

[WebGLShader](https://developer.mozilla.org/en/docs/Web/API/WebGLShader) to Kotlin
*/
public external abstract class WebGLShader : WebGLObject

/*
 * Exposes the JavaScript

[WebGLTexture](https://developer.mozilla.org/en/docs/Web/API/WebGLTexture) to Kotlin
*/
public external abstract class WebGLTexture : WebGLObject

/*
 * Exposes the JavaScript

[WebGLUniformLocation](https://developer.mozilla.org/en/docs/Web/API/WebGLUniformLocation) to Kotlin
*/
public external abstract class WebGLUniformLocation

/*
 * Exposes the JavaScript

[WebGLActiveInfo](https://developer.mozilla.org/en/docs/Web/API/WebGLActiveInfo) to Kotlin
*/
public external abstract class WebGLActiveInfo {
  open val size: Int
  open val type: Int
  open val name: String
}

/*
 * Exposes the JavaScript

[WebGLShaderPrecisionFormat](https://developer.mozilla.org/en/docs/Web/API/WebGLShaderPrecisionFormat) to Kotlin
*/
public external abstract class WebGLShaderPrecisionFormat {
  open val rangeMin: Int
  open val rangeMax: Int
}

@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface WebGLRenderingContextBase {
  val canvas: HTMLCanvasElement
  val drawingBufferWidth: Int
  val drawingBufferHeight: Int
  fun getContextAttributes(): WebGLContextAttributes?
  fun isContextLost(): Boolean
  fun getSupportedExtensions(): Array<String>?
  fun getExtension(name: String): dynamic
  fun activeTexture(texture: Int)
  fun attachShader(program: WebGLProgram?, shader: WebGLShader?)
  fun bindAttribLocation(program: WebGLProgram?, index: Int, name: String)
  fun bindBuffer(target: Int, buffer: WebGLBuffer?)
  fun bindFramebuffer(target: Int, framebuffer: WebGLFramebuffer?)
  fun bindRenderbuffer(target: Int, renderbuffer: WebGLRenderbuffer?)
  fun bindTexture(target: Int, texture: WebGLTexture?)
  fun blendColor(red: Float, green: Float, blue: Float, alpha: Float)
  fun blendEquation(mode: Int)
  fun blendEquationSeparate(modeRGB: Int, modeAlpha: Int)
  fun blendFunc(sfactor: Int, dfactor: Int)
  fun blendFuncSeparate(srcRGB: Int, dstRGB: Int, srcAlpha: Int, dstAlpha: Int)
  fun bufferData(target: Int, size: Int, usage: Int)
  fun bufferData(target: Int, data: BufferDataSource?, usage: Int)
  fun bufferSubData(target: Int, offset: Int, data: BufferDataSource?)
  fun checkFramebufferStatus(target: Int)
  fun clearColor(red: Float, green: Float, blue: Float, alpha: Float)
  fun clearDepth(depth: Float)
  fun clearStencil(s: Int)
  fun clearColor(red: Boolean, green: Boolean, blue: Boolean, alpha: Boolean)
  fun compileShader(shader: WebGLShader?)
  fun createBuffer(): WebGLBuffer?
  fun createFramebuffer(): WebGLFramebuffer?
  fun createProgram(): WebGLProgram?
fun createShader(type: Int): WebGLShader?
fun createTexture(): WebGLTexture?
fun cullFace(mode: Int)
fun deleteBuffer(buffer: WebGLBuffer?)
fun deleteFramebuffer(framebuffer: WebGLFramebuffer?)
fun deleteProgram(program: WebGLProgram?)
fun deleteRendererBuffer(rendererBuffer: WebGLRendererBuffer?)
fun deleteShader(shader: WebGLShader?)
fun deleteTexture(texture: WebGLTexture?)
fun depthFunc(func: Int)
fun depthMask(flag: Boolean)
fun depthRange(zNear: Float, zFar: Float)
fun detachShader(program: WebGLProgram?, shader: WebGLShader?)
fun disable(cap: Int)
fun disableVertexAttribArray(index: Int)
fun drawArrays(mode: Int, first: Int, count: Int)
fun drawElements(mode: Int, count: Int, type: Int, offset: Int)
fun enable(cap: Int)
fun enableVertexAttribArray(index: Int)
fun finish()
fun flush()
fun framebufferRenderbuffer(target: Int, renderbuffer: WebGLRenderbuffer?)
fun frontFace(mode: Int)
fun generateMipmap(target: Int)
fun getActiveAttrib(program: WebGLProgram?, index: Int): WebGLActiveInfo?
fun getActiveUniform(program: WebGLProgram?, index: Int): WebGLActiveInfo?
fun getAttachedShaders(program: WebGLProgram?): Array<WebGLShader>?
fun getAttribLocation(program: WebGLProgram?, name: String): Int
fun getBufferParameter(target: Int, pname: Int): Any?
fun getParameter(pname: Int): Any?
fun getError(): Int
fun getFramebufferAttachmentParameter(target: Int, attachment: Int, pname: Int): Any?
fun getProgramParameter(pname: Int): Any?
fun getProgramInfoLog(program: WebGLProgram?): String?
fun getProgramParameters(program: WebGLProgram?): Int
fun getRenderbufferParameter(target: Int, pname: Int): Any?
fun getShaderParameter(shader: WebGLShader?, pname: Int): Any?
fun getShaderPrecisionFormat(shaderType: Int, precisionType: Int): WebGLShaderPrecisionFormat?
fun getShaderInfoLog(shader: WebGLShader?): String?
fun getShaderSource(shader: WebGLShader?): String?
fun getTexParameter(target: Int, pname: Int): Any?
fun getUniformLocation(program: WebGLProgram?, name: String): WebGLUniformLocation?
fun getVertexAttrib(index: Int, pname: Int): Any?
fun getVertexAttribOffset(index: Int, pname: Int): Int
fun hint(target: Int, mode: Int)
fun isBuffer(buffer: WebGLBuffer?): Boolean
fun isEnabled(cap: Int): Boolean
fun isFramebuffer(framebuffer: WebGLFramebuffer?): Boolean
fun isProgram(program: WebGLProgram?): Boolean
fun isRendererBuffer(rendererBuffer: WebGLRendererBuffer?): Boolean
fun isShader(shader: WebGLShader?): Boolean
fun isTexture(texture: WebGLTexture?): Boolean
fun lineWidth(width: Float)
fun linkProgram(program: WebGLProgram?)
fun pixelStorei(pname: Int, param: Int)
fun polygonOffset(factor: Float, units: Float)
fun readPixels(x: Int, y: Int, width: Int, height: Int, format: Int, type: Int, pixels: ArrayBufferView?)
fun renderbufferStorage(target: Int, internalformat: Int, width: Int, height: Int)
fun sampleCoverage(value: Float, invert: Boolean)
fun scissor(x: Int, y: Int, width: Int, height: Int)
fun shaderSource(shader: WebGLShader?, source: String)
fun stencilFunc(func: Int, ref: Int, mask: Int)
fun stencilFuncSeparate(face: Int, ref: Int, mask: Int)
fun stencilMask(mask: Int)
fun stencilMaskSeparate(face: Int, mask: Int)
fun stencilOpFail(fail: Int, zfail: Int, zpass: Int)
fun texImage2D(target: Int, level: Int, internalformat: Int, format: Int, type: Int, pixels: ArrayBufferView?)
fun texImage2D(target: Int, level: Int, internalformat: Int, format: Int, type: Int, source:TexImageSource?)
fun texParameterf(target: Int, pname: Int, param: Float)
fun texParameteri(target: Int, pname: Int, param: Int)
fun texSubImage2D(target: Int, level: Int, xoffset: Int, yoffset: Int)
fun uniform1f(location: WebGLUniformLocation?, x: Float)
fun uniform1fv(location: WebGLUniformLocation?, v: Float32Array)
fun uniform1iv(location: WebGLUniformLocation?, v: Int32Array)
fun uniform2f(location: WebGLUniformLocation?, x: Float, y: Float)
fun uniform2fv(location: WebGLUniformLocation?, v: Float32Array)
fun uniform2iv(location: WebGLUniformLocation?, v: Int32Array)
val CLAMP_TO_EDGE: Int
val MIRRORED_REPEAT: Int
val FLOAT_VEC2: Int
val FLOAT_VEC3: Int
val FLOAT_VEC4: Int
val INT_VEC2: Int
val INT_VEC3: Int
val INT_VEC4: Int
val BOOL: Int
val BOOL_VEC2: Int
val BOOL_VEC3: Int
val BOOL_VEC4: Int
val FLOAT_MAT2: Int
val FLOAT_MAT3: Int
val FLOAT_MAT4: Int
val SAMPLER_2D: Int
val SAMPLER_CUBE: Int
val VERTEX_ATTRIB_ARRAY_ENABLED: Int
val VERTEX_ATTRIB_ARRAY_SIZE: Int
val VERTEX_ATTRIB_ARRAY_STRIDE: Int
val VERTEX_ATTRIB_ARRAY_TYPE: Int
val VERTEX_ATTRIB_ARRAY_NORMALIZED: Int
val VERTEX_ATTRIB_ARRAY_POINTER: Int
val VERTEX_ATTRIB_ARRAY_BUFFER_BINDING: Int
val IMPLEMENTATION_COLOR_READ_TYPE: Int
val IMPLEMENTATION_COLOR_READ_FORMAT: Int
val LOW_FLOAT: Int
val MEDIUM_FLOAT: Int
val HIGH_FLOAT: Int
val LOW_INT: Int
val MEDIUM_INT: Int
val HIGH_INT: Int
val FRAMEBUFFER: Int
val RENDERBUFFER: Int
val RGBA4: Int
val RGB5_A1: Int
val RGB565: Int
val DEPTH_COMPONENT16: Int
val STENCIL_INDEX: Int
val STENCIL_INDEX8: Int
val DEPTH_STENCIL: Int
val RENDERBUFFER_WIDTH: Int
val RENDERBUFFER_HEIGHT: Int
val RENDERBUFFER_INTERNAL_FORMAT: Int
val RENDERBUFFER_RED_SIZE: Int
val RENDERBUFFER_GREEN_SIZE: Int
val RENDERBUFFER_ALPHA_SIZE: Int
val RENDERBUFFER_DEPTH_SIZE: Int
val RENDERBUFFER_STENCIL_SIZE: Int
val FRAMEBUFFER_ATTACHMENT_OBJECT_TYPE: Int
val FRAMEBUFFER_ATTACHMENT_OBJECT_NAME: Int
val FRAMEBUFFER_ATTACHMENT_TEXTURE_LEVEL: Int
val FRAMEBUFFER_ATTACHMENT_TEXTURE_CUBE_MAP_FACE: Int
val COLOR_ATTACHMENT0: Int
val DEPTH_ATTACHMENT: Int
val STENCIL_ATTACHMENT: Int
val DEPTH_STENCIL_ATTACHMENT: Int
val NONE: Int
val FRAMEBUFFER_COMPLETE: Int
val FRAMEBUFFER_INCOMPLETE_ATTACHMENT: Int
val FRAMEBUFFER_INCOMPLETE_MISSING_ATTACHMENT: Int
val FRAMEBUFFER_INCOMPLETE_DIMENSIONS: Int
val FRAMEBUFFER_UNSUPPORTED: Int
val FRAMEBUFFER_BINDING: Int
val RENDERBUFFER_BINDING: Int
val MAX_RENDERBUFFER_SIZE: Int
val INVALID_FRAMEBUFFER_OPERATION: Int
val UNPACK_FLIP_Y_WEBGL: Int
val UNPACK_PREMULTIPLY_ALPHA_WEBGL: Int
val CONTEXT_LOST_WEBGL: Int
val BROWSER_DEFAULT_WEBGL: Int

/**
 * Exposes the JavaScript [WebGLRenderingContext](https://developer.mozilla.org/en/docs/Web/API/WebGLRenderingContext) to Kotlin
 */
public external abstract class WebGLRenderingContext : WebGLRenderingContextBase, RenderingContext {

companion object {
val DEPTH_BUFFER_BIT: Int
val STENCIL_BUFFER_BIT: Int
val COLOR_BUFFER_BIT: Int
val POINTS: Int
val LINES: Int
val LINE_LOOP: Int
val LINE_STRIP: Int
val TRIANGLES: Int
val TRIANGLE_STRIP: Int
val TRIANGLE_FAN: Int
val ZERO: Int
val ONE: Int
val SRC_COLOR: Int
val ONE_MINUS_SRC_COLOR: Int
val SRC_ALPHA: Int
val ONE_MINUS_SRC_ALPHA: Int
val DST_ALPHA: Int
val ONE_MINUS_DST_ALPHA: Int
val SRC_ALPHA_SATURATE: Int
val FUNC_ADD: Int
val BLEND_EQUATION: Int
val BLEND_EQUATION_RGB: Int
val BLEND_EQUATION_ALPHA: Int
val FUNC_SUBTRACT: Int
val BLEND_SRC_RGB: Int
val BLEND_DST_RGB: Int
val BLEND_SRC_ALPHA: Int
val BLEND_DST_ALPHA: Int
val CONSTANT_COLOR: Int
val ONE_MINUS_CONSTANT_COLOR: Int
val CONSTANT_ALPHA: Int
val ONE_MINUS_CONSTANT_ALPHA: Int
val ARRAY_BUFFER: Int
val ELEMENT_ARRAY_BUFFER: Int
val ARRAY_BUFFER_BINDING: Int
val ELEMENT_ARRAY_BUFFER_BINDING: Int
val STREAM_DRAW: Int
val
TEXTURE_CUBE_MAP_POSITIVE_Y: Int
        val TEXTURE_CUBE_MAP_NEGATIVE_Y: Int
        val MAX_CUBE_MAP_TEXTURE_SIZE: Int
        val TEXTURE0: Int
        val TEXTURE1: Int
        val TEXTURE2: Int
        val TEXTURE3: Int
        val TEXTURE4: Int
        val TEXTURE5: Int
        val TEXTURE6: Int
        val TEXTURE7: Int
        val TEXTURE8: Int
        val TEXTURE9: Int
        val TEXTURE10: Int
        val TEXTURE11: Int
        val TEXTURE12: Int
        val TEXTURE13: Int
        val TEXTURE14: Int
        val TEXTURE15: Int
        val TEXTURE16: Int
        val TEXTURE17: Int
        val TEXTURE18: Int
        val TEXTURE19: Int
        val TEXTURE20: Int
        val TEXTURE21: Int
        val TEXTURE22: Int
        val TEXTURE23: Int
        val TEXTURE24: Int
        val TEXTURE25: Int
        val TEXTURE26: Int
        val TEXTURE27: Int
        val TEXTURE28: Int
        val TEXTURE29: Int
        val TEXTURE30: Int
        val TEXTURE31: Int
        val ACTIVE_TEXTURE: Int
        val REPEAT: Int
        val CLAMP_TO_EDGE: Int
        val MIRRORED_REPEAT: Int
        val FLOAT_VEC2: Int
        val FLOAT_VEC3: Int
        val FLOAT_VEC4: Int
        val INT_VEC2: Int
        val INT_VEC3: Int
        val INT_VEC4: Int
        val BOOL: Int
        val BOOL_VEC2: Int
        val BOOL_VEC3: Int
        val BOOL_VEC4: Int
        val FLOAT_MAT2: Int
        val FLOAT_MAT3: Int
        val FLOAT_MAT4: Int
        val SAMPLER_2D: Int
        val SAMPLER_CUBE: Int
        val VERTEX_ATTRIB_ARRAY_ENABLED: Int
        val VERTEX_ATTRIB_ARRAY_SIZE: Int
        val VERTEX_ATTRIB_ARRAY_STRIDE: Int
        val VERTEX_ATTRIB_ARRAY_TYPE: Int
        val VERTEX_ATTRIB_ARRAY_NORMALIZED: Int
        val VERTEX_ATTRIB_ARRAY_POINTER: Int
        val VERTEX_ATTRIB_ARRAY_BUFFER_BINDING: Int
        val IMPLEMENTATION_COLOR_READ_TYPE: Int
        val IMPLEMENTATION_COLOR_READ_FORMAT: Int
        val LOW_FLOAT: Int
        val MEDIUM_FLOAT: Int
        val HIGH_FLOAT: Int
        val LOW_INT: Int
        val MEDIUM_INT: Int
        val HIGH_INT: Int
        val FRAMEBUFFER: Int
        val RENDERBUFFER: Int
        val RGBA4: Int
        val RGB5_A1: Int
        val RGB565: Int
        val DEPTH_COMPONENT16: Int
        val STENCIL_INDEX: Int
        val STENCIL_INDEX8: Int
        val DEPTH_STENCIL: Int
        val RENDERBUFFER_WIDTH: Int
        val RENDERBUFFER_HEIGHT: Int
        val RENDERBUFFER_INTERNAL_FORMAT: Int
        val RENDERBUFFER_RED_SIZE: Int
        val RENDERBUFFER_GREEN_SIZE: Int
        val RENDERBUFFER_BLUE_SIZE: Int
        val RENDERBUFFER_ALPHA_SIZE: Int
        val RENDERBUFFER_DEPTH_SIZE: Int
        val RENDERBUFFER_STENCIL_SIZE: Int
        val FRAMEBUFFER_ATTACHMENT_OBJECT_TYPE: Int
        val FRAMEBUFFER_ATTACHMENT_OBJECT_NAME: Int
        val FRAMEBUFFER_ATTACHMENT_TEXTURE_LEVEL: Int
        val FRAMEBUFFER_ATTACHMENT_TEXTURE_CUBE_MAP_FACE: Int
        val COLOR_ATTACHMENT0: Int
        val DEPTH_ATTACHMENT: Int
        val STENCIL_ATTACHMENT: Int
        val NONE: Int
        val FRAMEBUFFER_COMPLETE: Int
        val FRAMEBUFFER_INCOMPLETE_ATTACHMENT: Int
        val FRAMEBUFFER_INCOMPLETE_MISSING_ATTACHMENT: Int
        val FRAMEBUFFER_UNSUPPORTED: Int
        val FRAMEBUFFER_BINDING: Int
        val RENDERBUFFER_BINDING: Int
        val MAX_RENDERBUFFER_SIZE: Int
        val INVALID_FRAMEBUFFER_OPERATION: Int
        val UNPACK_FLIP_Y_WEBGL: Int
        val UNPACK_PREMULTIPLY_ALPHA_WEBGL: Int
        val CONTEXT_LOST_WEBGL: Int
        val BROWSER_DEFAULT_WEBGL: Int
        val UNPACK_CODING_WEBGL: Int
        val CONTEXT_LOST_WEBGL: Int
        val CONTEXT_TRACE_MODE_WEBGL: Int
        val CONTEXT_FENCE_WEBGL: Int
        val CONTEXT_INQUIRY_WEBGL: Int
        val CONTEXT_PROPERTIES_WEBGL: Int
        val CONTEXT_LOST_WEBGL: Int
        val CONTEXT_TRACE_MODE_WEBGL: Int
        val CONTEXT_FENCE_WEBGL: Int
        val CONTEXT_INQUIRY_WEBGL: Int
        val CONTEXT_PROPERTIES_WEBGL: Int

    }
}

/**
 * Exposes the JavaScript [WebGLContextEvent](https://developer.mozilla.org/en/docs/Web/API/WebGLContextEvent) to Kotlin
 */
public external open class WebGLContextEvent(type: String, eventInit: WebGLContextEventInit = null) : Event {
    open val statusMessage: String

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
        val AT_TARGET: Short
    }

public external interface WebGLContextEventInit : EventInit {
    var statusMessage: String? /* = "" */

    fun definedExternally()}

Open Source Used In security_automation identity-guest-cs 1646
fun Int16Array.set(index: Int, value: Short) { asDynamic()[index] = value }

/**
 * Exposes the JavaScript [Int16Array](https://developer.mozilla.org/en/docs/Web/API/Int16Array) to Kotlin.*n/public external open class

Int16Array : ArrayBufferView {
    constructor(array: Uint16Array)
    constructor(array: Array<Short>)
    constructor(buffer: ArrayBuffer, byteOffset: Int = definedExternally, length: Int = definedExternally) open val length: Int
    override fun set(array: Uint16Array, offset: Int)
    override fun set(array: Array<Short>, offset: Int = definedExternally)
    fun subarray(start: Int, end: Int): Uint16Array

    companion object {
        val BYTES_PER_ELEMENT: Int
    }
}

@kotlin.internal.InlineOnly
public inline fun Uint16Array.get(index: Int): Short = asDynamic()[index]

@kotlin.internal.InlineOnly
public inline fun Uint16Array.set(index: Int, value: Short) { asDynamic()[index] = value }

/**
 * Exposes the JavaScript [Uint16Array](https://developer.mozilla.org/en/docs/Web/API/Uint16Array) to Kotlin.*n/public external open class

Uint16Array : ArrayBufferView {
    constructor(array: Uint16Array)
    constructor(array: Array<Short>)
    constructor(buffer: ArrayBuffer, byteOffset: Int = definedExternally, length: Int = definedExternally) open val byteLength: Int
    override fun set(array: Uint16Array, offset: Int)
    override fun set(array: Array<Short>, offset: Int = definedExternally)
    fun subarray(start: Int, end: Int): Uint16Array

    companion object {
        val BYTES_PER_ELEMENT: Int
    }
}

@kotlin.internal.InlineOnly
public inline fun Uint16Array.get(index: Int): Int = asDynamic()[index]

@kotlin.internal.InlineOnly
public inline fun Uint16Array.set(index: Int, value: Int) { asDynamic()[index] = value }

/**
 * Exposes the JavaScript [Int32Array](https://developer.mozilla.org/en/docs/Web/API/Int32Array) to Kotlin.*n/public external open class

Int32Array : ArrayBufferView {
    constructor(array: Int32Array)
    constructor(array: Array<Int>)
    constructor(buffer: ArrayBuffer, byteOffset: Int = definedExternally, length: Int = definedExternally)
    open val length: Int
    override val buffer: ArrayBuffer
    override val byteOffset: Int
    override val byteLength: Int
    fun set(array: Int32Array, offset: Int = definedExternally)
    fun set(array: Array<Int>, offset: Int = definedExternally)
    fun subarray(start: Int, end: Int): Int32Array

    companion object {
        val BYTES_PER_ELEMENT: Int
    }
}

@kotlin.internal.InlineOnly
public inline fun Int32Array.get(index: Int): Int = asDynamic()[index]

@kotlin.internal.InlineOnly
public inline fun Int32Array.set(index: Int, value: Int) { asDynamic()[index] = value }

/**
 * Exposes the JavaScript [Uint32Array](https://developer.mozilla.org/en/docs/Web/API/Uint32Array) to Kotlin.*n/public external open class

Uint32Array : ArrayBufferView {
    constructor(array: Uint32Array)
    constructor(array: Array<Int>)
    constructor(buffer: ArrayBuffer, byteOffset: Int = definedExternally, length: Int = definedExternally)
    open val length: Int
    override val buffer: ArrayBuffer
    override val byteOffset: Int
    override val byteLength: Int
    fun set(array: Uint32Array, offset: Int = definedExternally)
    fun set(array: Array<Int>, offset: Int = definedExternally)
    fun subarray(start: Int, end: Int): Uint32Array

    companion object {
        val BYTES_PER_ELEMENT: Int
    }
}

@kotlin.internal.InlineOnly
public inline fun Uint32Array.get(index: Int): Int = asDynamic()[index]

@kotlin.internal.InlineOnly
public inline fun Uint32Array.set(index: Int, value: Int) { asDynamic()[index] = value }

/**
 * Exposes the JavaScript [Float32Array](https://developer.mozilla.org/en/docs/Web/API/Float32Array) to Kotlin.*n/public external open class

Float32Array : ArrayBufferView {
    constructor(array: Float32Array)
    constructor(array: Array<Float>)
    constructor(buffer: ArrayBuffer, byteOffset: Int = definedExternally, length: Int = definedExternally)
    open val length: Int
    override val buffer: ArrayBuffer
    override val byteOffset: Int
    override val byteLength: Int
    fun set(array: Float32Array, offset: Int = definedExternally)
    fun set(array: Array<Float>, offset: Int = definedExternally)
    fun subarray(start: Int, end: Int): Float32Array

    companion object {
        val BYTES_PER_ELEMENT: Int
    }
}

@kotlin.internal.InlineOnly
public inline fun Float32Array.get(index: Int): Float = asDynamic()[index]

@kotlin.internal.InlineOnly
public inline fun Float32Array.set(index: Int, value: Float) { asDynamic()[index] = value }

/**
 * Exposes the JavaScript [Float64Array](https://developer.mozilla.org/en/docs/Web/API/Float64Array) to Kotlin.*n/public external open class

Float64Array : ArrayBufferView {
    constructor(array: Float64Array)
    constructor(array: Array<Double>)
    constructor(buffer: ArrayBuffer, byteOffset: Int = definedExternally, length: Int = definedExternally)
    open val length: Int
    override val buffer: ArrayBuffer
    override val byteOffset: Int
    override val byteLength: Int
    fun set(array: Float64Array, offset: Int = definedExternally)
    fun set(array: Array<Double>, offset: Int = definedExternally)
    fun subarray(start: Int, end: Int): Float64Array

    companion object {
        val BYTES_PER_ELEMENT: Int
    }
}

@kotlin.internal.InlineOnly
public inline fun Float64Array.get(index: Int): Double = asDynamic()[index]

@kotlin.internal.InlineOnly
public inline fun Float64Array.set(index: Int, value: Double) { asDynamic()[index] = value }

/**
 * Exposes the JavaScript [DataView](https://developer.mozilla.org/en/docs/Web/API/DataView) to Kotlin.*n/public external open class

DataView : ArrayBufferView {
    constructor(array: DataView)
    constructor(array: Array<Double>)
    constructor(buffer: ArrayBuffer, byteOffset: Int = definedExternally, length: Int = definedExternally)
    open val length: Int
    override val buffer: ArrayBuffer
    override val byteOffset: Int
    override val byteLength: Int
    fun set(array: DataView, offset: Int = definedExternally)
    fun set(array: Array<Double>, offset: Int = definedExternally)
    fun subarray(start: Int, end: Int): DataView

    companion object {
        val BYTES_PER_ELEMENT: Int
    }
}

@kotlin.internal.InlineOnly
public inline fun DataView.get(index: Int): Double = asDynamic()[index]

@kotlin.internal.InlineOnly
public inline fun DataView.set(index: Int, value: Double) { asDynamic()[index] = value }
**Open Source Used In security Automation identity-guest-cs**

---

**DataView**(buffer: ArrayBuffer, byteOffset: Int = definedExternally, byteLength: Int = definedExternally) : ArrayBufferView

- fun getInt8(byteOffset: Int): Byte
- fun getInt16(byteOffset: Int, littleEndian: Boolean = definedExternally): Short
- fun getInt32(byteOffset: Int, littleEndian: Boolean = definedExternally): Int
- fun getFloat32(byteOffset: Int, littleEndian: Boolean = definedExternally): Float
- fun getFloat64(byteOffset: Int, littleEndian: Boolean = definedExternally): Double

- fun setInt8(byteOffset: Int, value: Byte)
- fun setUint8(byteOffset: Int, value: Byte)
- fun setInt16(byteOffset: Int, value: Short, littleEndian: Boolean = definedExternally)
- fun setUint16(byteOffset: Int, value: Short, littleEndian: Boolean = definedExternally)
- fun setInt32(byteOffset: Int, value: Int, littleEndian: Boolean = definedExternally)
- fun setUint32(byteOffset: Int, value: Int, littleEndian: Boolean = definedExternally)
- fun setFloat32(byteOffset: Int, value: Float, littleEndian: Boolean = definedExternally)
- fun setFloat64(byteOffset: Int, value: Double, littleEndian: Boolean = definedExternally)

---

**BufferDataSource**

---

**TexImageSource**

---

/*
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

// NOTE: THIS FILE IS AUTO-GENERATED, DO NOT EDIT!/n// See github.com/kotlin/dukat for details

---

package org.w3c.dom.clipboard

import kotlin.js.*
import org.khronos.webgl.*
import org.w3c.css.masking.*
import org.w3c.dom.*
import org.w3c.dom.css.*
import org.w3c.dom.encryptedmedia.*
import org.w3c.dom.events.*
import org.w3c.dom.mediacapture.*
import org.w3c.dom.mediasource.*
import org.w3c.dom.parsing.*
import org.w3c.dom.pointerevents.*
import org.w3c.dom.svg.*
import org.w3c.dom.url.*
import org.w3c.fetch.*
import org.w3c.files.*
import org.w3c.notifications.*
import org.w3c.performance.*
import org.w3c.workers.*
import org.w3c.xhr.*

public external interface ClipboardEventInit : EventInit {
    var clipboardData: DataTransfer? /* = null */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["clipboardData"] = clipboardData
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}

/**
 * Exposes the JavaScript

---

<table>
<thead>
<tr>
<th>ClipboardEvent</th>
<th>Clipboard</th>
<th>ClipboardPermissionDescriptor</th>
</tr>
</thead>
<tbody>
<tr>
<td>(data: DataTransfer)</td>
<td>Promise&lt;Unit&gt;</td>
<td></td>
</tr>
<tr>
<td>(data: String)</td>
<td>Promise&lt;Unit&gt;</td>
<td></td>
</tr>
<tr>
<td>(allowWithoutGesture: Boolean? = false, allowWithoutGesture: Boolean? = false)</td>
<td>Promise&lt;Unit&gt;</td>
<td></td>
</tr>
</tbody>
</table>
org.w3c.fetch.
import org.w3c.files.
import org.w3c.notifications.
import org.w3c.performance.
import org.w3c.workers.

public external abstract class MediaList : ItemArrayLike<String> {
    open var mediaText: String
    fun appendMedium(medium: String)
    fun deleteMedium(medium: String)

    override fun item(index: Int): String?
}

@kotlin.internal.InlineOnly
public inline operator fun MediaList.get(index: Int): String? = asDynamic()[index]

/**
 * Exposes the JavaScript [StyleSheet](https://developer.mozilla.org/en/docs/Web/API/StyleSheet) to Kotlin
 */
public external abstract class StyleSheet {
    open val type: String
    open val href: String?
    open val ownerNode: UnionElementOrProcessingInstruction
    open val parentStyleSheet: StyleSheet
    open val title: String?

    open val media: MediaList
    open var disabled: Boolean
}

/**
 * Exposes the JavaScript [CSSStyleSheet](https://developer.mozilla.org/en/docs/Web/API/CSSStyleSheet) to Kotlin
 */
public external abstract class CSSStyleSheet : StyleSheet {
    open val ownerRule: CSSRule?
    open val cssRules: CSSRuleList
    fun insertRule(rule: String, index: Int): Int
    fun deleteRule(index: Int)
}

/**
 * Exposes the JavaScript [StyleSheetList](https://developer.mozilla.org/en/docs/Web/API/StyleSheetList) to Kotlin
 */
public external abstract class StyleSheetList : ItemArrayLike<StyleSheet> {
    override fun item(index: Int): StyleSheet?
}

@kotlin.internal.InlineOnly
public inline operator fun StyleSheetList.get(index: Int): StyleSheet? = asDynamic()[index]

/**
 * Exposes the JavaScript [LinkStyle](https://developer.mozilla.org/en/docs/Web/API/LinkStyle) to Kotlin
 */
public external interface LinkStyle {
    val sheet: StyleSheet
        get() = definedExternally
}

/**
 * Exposes the JavaScript [CSSRuleList](https://developer.mozilla.org/en/docs/Web/API/CSSRuleList) to Kotlin
 */
public external abstract class CSSRuleList : ItemArrayLike<CSSRule> {
    override fun item(index: Int): CSSRule?
}

@kotlin.internal.InlineOnly
public inline operator fun CSSRuleList.get(index: Int): CSSRule? = asDynamic()[index]

/**
 * Exposes the JavaScript [CSSRule](https://developer.mozilla.org/en/docs/Web/API/CSSRule) to Kotlin
 */
public external abstract class CSSRule {
    open val type: Short
    open var cssText: String
    open val parentRule: CSSRule?
    open val parentStyleSheet: CSSStyleSheet?

    companion object {
        val STYLE_RULE: Short
        val CHARSET_RULE: Short
        val IMPORT_RULE: Short
        val MEDIA_RULE: Short
        val FONT_FACE_RULE: Short
        val PAGE_RULE: Short
        val MARGIN_RULE: Short
        val NAMESPACE_RULE: Short
    }
}

/**
 * Exposes the JavaScript [CSSStyleRule](https://developer.mozilla.org/en/docs/Web/API/CSSStyleRule) to Kotlin
 */
public external abstract class CSSStyleRule : CSSRule {
    open var selectorText: String
    open val style: CSSStyleDeclaration

    companion object {
        val STYLE_RULE: Short
        val CHARSET_RULE: Short
        val IMPORT_RULE: Short
        val MEDIA_RULE: Short
        val FONT_FACE_RULE: Short
        val PAGE_RULE: Short
        val MARGIN_RULE: Short
        val NAMESPACE_RULE: Short
    }
}

/**
 * Exposes the JavaScript [CSSImportRule](https://developer.mozilla.org/en/docs/Web/API/CSSImportRule) to Kotlin
 */
public external abstract class CSSImportRule : CSSRule {
    open val href: String
    open val media: MediaList
    open val styleSheet: CSSStyleSheet

    companion object {
        val STYLE_RULE: Short
        val CHARSET_RULE: Short
        val IMPORT_RULE: Short
        val MEDIA_RULE: Short
        val FONT_FACE_RULE: Short
        val PAGE_RULE: Short
        val MARGIN_RULE: Short
        val NAMESPACE_RULE: Short
    }
}

/**
 * Exposes the JavaScript [CSSGroupingRule](https://developer.mozilla.org/en/docs/Web/API/CSSGroupingRule) to Kotlin
 */
public external abstract class CSSGroupingRule : CSSRule {
    open val cssRules: CSSRuleList
    fun insertRule(rule: String, index: Int): Int
    fun deleteRule(index: Int)

    companion object {
        val STYLE_RULE: Short
        val CHARSET_RULE: Short
        val IMPORT_RULE: Short
        val MEDIA_RULE: Short
        val FONT_FACE_RULE: Short
        val PAGE_RULE: Short
        val MARGIN_RULE: Short
        val NAMESPACE_RULE: Short
    }
}

/**
 * Exposes the JavaScript [CSSMediaRule](https://developer.mozilla.org/en/docs/Web/API/CSSMediaRule) to Kotlin
 */
public external abstract class CSSMediaRule : CSSGroupingRule {
    open val media: MediaList

    companion object {
        val STYLE_RULE: Short
        val CHARSET_RULE: Short
        val IMPORT_RULE: Short
        val MEDIA_RULE: Short
        val FONT_FACE_RULE: Short
        val PAGE_RULE: Short
        val MARGIN_RULE: Short
        val NAMESPACE_RULE: Short
    }
}
/**
 * Exposes the JavaScript [CSSPageRule](https://developer.mozilla.org/en/docs/Web/API/CSSPageRule) to Kotlin
 */
public external abstract class CSSPageRule : CSSGroupingRule {
    open var selectorText: String
    open val style: CSSStyleDeclaration

    companion object {
        val STYLE_RULE: Short
        val CHARSET_RULE: Short
        val IMPORT_RULE: Short
        val MEDIA_RULE: Short
        val FONT_FACE_RULE: Short
        val PAGE_RULE: Short
        val MARGIN_RULE: Short
        val NAMESPACE_RULE: Short
    }
}

/**
 * Exposes the JavaScript [CSSMarginRule](https://developer.mozilla.org/en/docs/Web/API/CSSMarginRule) to Kotlin
 */
public external abstract class CSSMarginRule : CSSRule {
    open val name: String
    open val style: CSSStyleDeclaration

    companion object {
        val STYLE_RULE: Short
        val CHARSET_RULE: Short
        val IMPORT_RULE: Short
        val MEDIA_RULE: Short
        val FONT_FACE_RULE: Short
        val PAGE_RULE: Short
        val MARGIN_RULE: Short
        val NAMESPACE_RULE: Short
    }
}

/**
 * Exposes the JavaScript [CSSNamespaceRule](https://developer.mozilla.org/en/docs/Web/API/CSSNamespaceRule) to Kotlin
 */
public external abstract class CSSNamespaceRule : CSSRule {
    open val namespaceURI: String
    open val prefix: String

    companion object {
        val STYLE_RULE: Short
        val CHARSET_RULE: Short
        val IMPORT_RULE: Short
        val MEDIA_RULE: Short
        val FONT_FACE_RULE: Short
        val PAGE_RULE: Short
        val MARGIN_RULE: Short
        val NAMESPACE_RULE: Short
    }
}

/**
 * Exposes the JavaScript [CSSStyleDeclaration](https://developer.mozilla.org/en/docs/Web/API/CSSStyleDeclaration) to Kotlin
 */
public external abstract class CSSStyleDeclaration : ItemArrayLike<String> {
    open var cssText: String
    open var parentRule: CSSRule?
    open var cssFloat: String
    open var alignItems: String
    open var alignSelf: String
    open var animation: String
    open var animationDelay: String
    open var animationDirection: String
    open var animationDuration: String
    open var animationFillMode: String
    open var animationIterationCount: String
    open var animationName: String
    open var animationPlayState: String
    open var animationTimingFunction: String
    open var background: String
    open var backgroundAttachment: String
    open var backgroundColor: String
    open var backgroundImage: String
    open var backgroundOrigin: String
    open var backgroundPosition: String
    open var backgroundRepeat: String
    open var backgroundSize: String
    open var border: String
    open var borderBottom: String
    open var borderBottomColor: String
    open var borderBottomLeftRadius: String
    open var borderBottomRightRadius: String
    open var borderBottomStyle: String
    open var borderBottomWidth: String
    open var borderCollapse: String
    open var borderColor: String
    open var borderImage: String
    open var borderImageOutset: String
    open var borderImageRepeat: String
    open var borderImageSlice: String
    open var borderTop: String
    open var borderTopColor: String
    open var borderTopLeftRadius: String
    open var borderTopRightRadius: String
    open var borderTopStyle: String
    open var borderTopWidth: String
    open var borderWidth: String
    open var bottom: String
    open var boxDecorationBreak: String
    open var boxShadow: String
    open var boxSizing: String
    open var breakAfter: String
    open var breakBefore: String
    open var breakInside: String
    open var captionSide: String
    open var clear: String
    open var clip: String
    open var color: String
    open var columnCount: String
    open var columnFill: String
    open var columnGap: String
    open var columnRule: String
    open var columnRuleColor: String
    open var columnRuleStyle: String
    open var columnRuleWidth: String
    open var columnSpan: String
    open var columnWidth: String
    open var columns: String
    open var content: String
    open var counterIncrement: String
    open var counterReset: String
    open var cursor: String
    open var direction: String
    open var display: String
    open var emptyCells: String
    open var filter: String
    open var flex: String
    open var flexBasis: String
    open var flexGrow: String
    open var
public external abstract class CSS {
    companion object {
        fun escape(ident: String): String
    }
}

public external interface

package org.w3c.dom.encryptedmedia

import kotlin.js.*
import org.khronos.webgl.*
import org.w3c.dom.*
import org.w3c.dom.clipboard.*
import org.w3c.dom.css.*
import org.w3c.dom.events.*
import org.w3c.dom.mediacapture.*
import org.w3c.dom.mediasource.*
import org.w3c.dom.parsing.*
import org.w3c.dom.pointerevents.*
import org.w3c.dom.svg.*
import org.w3c.dom.url.*
import org.w3c.fetch.*
import org.w3c.files.*
import org.w3c.notifications.*
import org.w3c.performance.*
import org.w3c.workers.*
import org.w3c.xhr.*

public external interface MediaKeySystemConfiguration {
    var label: String? /* = "" */
        get() = definedExternally
        set(value) = definedExternally
    var initDataTypes: Array<String>? /* = arrayOf() */
        get() = definedExternally
        set(value) = definedExternally
    var audioCapabilities: Array<MediaKeySystemMediaCapability>? /* = arrayOf() */
        get() = definedExternally
        set(value) = definedExternally
    var videoCapabilities: Array<MediaKeySystemMediaCapability>? /* = arrayOf() */
        get() = definedExternally
        set(value) = definedExternally
    var distinctiveIdentifier: MediaKeysRequirement? /* = MediaKeysRequirement.OPTIONAL */
        get() = definedExternally
        set(value) = definedExternally
    var persistentState: MediaKeysRequirement? /* = MediaKeysRequirement.OPTIONAL */
        get() = definedExternally
        set(value) = definedExternally
    var sessionTypes: Array<String>?
        get() = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["label"] = label
    o["initDataTypes"] = initDataTypes
    o["audioCapabilities"] = audioCapabilities
    o["videoCapabilities"] = videoCapabilities
    o["distinctiveIdentifier"] = distinctiveIdentifier
    o["persistentState"] = persistentState
    o["sessionTypes"] = sessionTypes
    return o
}

public external interface MediaKeySystemMediaCapability {
    var contentType: String? /* = "" */
        get() = definedExternally
        set(value) = definedExternally
    var robustness: String? /* = "" */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun MediaKeySystemMediaCapability(contentType: String? = "", robustness: String? = ""): MediaKeySystemMediaCapability {
    val o = js("({})")
    o["contentType"] = contentType
    o["robustness"] = robustness
    return o
}

public external abstract class MediaKeySystemAccess {
    val keySystem: String
    fun getConfiguration(): MediaKeySystemConfiguration
    fun createMediaKeys(): Promise<MediaKeys>
}

public external abstract class MediaKeys {
    fun createSession(sessionType: MediaKeySessionType = definedExternally): MediaKeySession
}

public external abstract class MediaKeySession : EventTarget {
    open val sessionId: String
    open val expiration: Double
    open val closed: Promise<Unit>
    open val keyStatuses: MediaKeyStatusMap
    open var onkeystatuseschange: ((Event) -> dynamic)?
    open var onmessage: ((MessageEvent) -> dynamic)?
    fun generateRequest(initDataType: String, initData: dynamic): Promise<MediaKeys>
    fun load(sessionId: String): Promise<MediaKeySession>: EventTarget {
        val open val sessionId: String
        val open val expiration: Double
        val open val closed: Promise<Unit>
        val open val keyStatuses: MediaKeyStatusMap
        open val onkeystatuschange: ((Event) -> dynamic)?
        open val onmessage: ((MessageEvent) -> dynamic)?
        fun generateRequest(initDataType: String, initData: dynamic): Promise<MediaKeys>
        fun load(sessionId: String): Promise<MediaKeySession>: EventTarget {
            val open val sessionId: String
            val open val expiration: Double
            val open val closed: Promise<Unit>
        }
    }
}

public external open class MediaKeyStatusMap {
    open val size: Int
    fun has(keyId: dynamic): Boolean
    fun get(keyId: dynamic): Any
}

public external open class MediaKeyMessageEvent(type: String, eventInitDict: MediaKeyMessageEventInit): Event {
    val open val
error" as Dynamic().unsafeCast<MediaKeyStatus>()

/* please, don’t implement this interface!

@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface

public inline val MediaKeyMessageType.Companion.LICENSE_REQUEST: MediaKeyMessageType get() = "license-request" as Dynamic().unsafeCast<MediaKeyMessageType>()

public inline val MediaKeyMessageType.Companion.LICENSE_RENEWAL: MediaKeyMessageType get() = "license-renewal" as Dynamic().unsafeCast<MediaKeyMessageType>()

public inline val MediaKeyMessageType.Companion.LICENSE_RELEASE: MediaKeyMessageType get() = "license-release" as Dynamic().unsafeCast<MediaKeyMessageType>()

public inline val MediaKeyMessageType.Companion.INDIVIDUALIZATION_REQUEST: MediaKeyMessageType get() = "individualization-request" as Dynamic().unsafeCast<MediaKeyMessageType>()

/*
 * Copyright 2010-2020
 * JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

// NOTE: THIS FILE IS AUTO-GENERATED, DO NOT EDIT!
See github.com/kotlin/dukat for details

org.w3c.dom.events

import kotlin.js.*
import org.khronos.webgl.*
import org.w3c.css.masking.*
import org.w3c.dom.*
import org.w3c.dom.clipboard.*
import org.w3c.dom.mediacapture.*
import org.w3c.dom.mediasource.*
import org.w3c.dom.parsing.*
import org.w3c.dom.pointerevents.*
import org.w3c.dom.svg.*
import org.w3c.dom.url.*
import org.w3c.fetch.*
import org.w3c.files.*
import org.w3c.notifications.*
import org.w3c.performance.*
import org.w3c.workers.*
import org.w3c.xhr.*

/**
 * Exposes the JavaScript [UIEvent](https://developer.mozilla.org/en/docs/Web/API/UIEvent) to Kotlin
 */
public external open class UIEvent(type: String, eventInitDict: UIEventInit = definedExternally) : Event {
    open val view: Window?

    open val detail: Int

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }

    public external interface UIEventInit : EventInit {
        var view: Window? /* = null */
            get() = definedExternally
        set(value) = definedExternally
        var detail: Int? /* = 0 */
            get() = definedExternally
        set(value) = definedExternally
    }

    @kotlin.internal.InlineOnly
        val o = js("({})")
        o["view"] = view
        o["detail"] = detail
        o["bubbles"] = bubbles
        o["cancelable"] = cancelable
        o["composed"] = composed
        o["return"] = undefined
        return o
    }

    /**
     * Exposes the JavaScript [FocusEvent](https://developer.mozilla.org/en/docs/Web/API/FocusEvent) to Kotlin
     */
    public external open class FocusEvent(type: String, eventInitDict: FocusEventInit = definedExternally) : UIEvent {
        open val relatedTarget:

        companion object {
            val NONE: Short
            val CAPTURING_PHASE: Short
            val AT_TARGET: Short
            val BUBBLING_PHASE: Short
        }

        public external interface FocusEventInit : UIEventInit {
            var relatedTarget: EventTarget? /* = null */
                get() = definedExternally
        }

        @kotlin.internal.InlineOnly
            val o = js("({})")
            o["relatedTarget"] = relatedTarget
            o["view"] = view
            o["detail"] = detail
            o["bubbles"] = bubbles
            o["cancelable"] = cancelable
            return o
        }

        /**
         * Exposes the JavaScript [MouseEvent](https://developer.mozilla.org/en/docs/Web/API/MouseEvent) to Kotlin
         */
        public external open class MouseEvent(type: String, eventInitDict: MouseEventInit = definedExternally) : UIEvent, UnionElementOrMouseEvent {
            open val screenX: Int
            open val screenY: Int
            open val clientX: Int
            open val clientY: Int
            open val ctrlKey: Boolean
            open val shiftKey: Boolean
            open val altKey: Boolean
            open val metaKey: Boolean
            open val button: Short
            open val buttons: Short
            open val region: String
            fun getModifierState(keyArg: String): Boolean

            companion object {
                val NONE: Short
                val BUBBLING_PHASE: Short
                val CAPTURING_PHASE: Short
                val AT_TARGET: Short
            }

            @kotlin.internal.InlineOnly
            public inline fun MouseEventInit(screenX: Int, screenY: Int, clientX: Int, clientY: Int, ctrlKey: Boolean, shiftKey: Boolean, altKey: Boolean, metaKey: Boolean, button: Short, buttons: Short, region: String, getModifierState: Boolean): MouseEventInit {
                val o = js("({})")
                o["screenX"] = screenX
                o["screenY"] = screenY
                o["clientX"] = clientX
                o["clientY"] = clientY
                o["ctrlKey"] = ctrlKey
                o["shiftKey"] = shiftKey
                o["altKey"] = altKey
                o["metaKey"] = metaKey
                o["button"] = button
                o["buttons"] = buttons
                o["region"] = region
                o["getModifierState"] = getModifierState
                return o
            }
        }
    }
CAPTURING_PHASE: Short()
val AT_TARGET: Short()
val BUBBLING_PHASE: Short()

public external interface MouseEventInit : EventModifierInit {
    var screenX: Int? /* = 0 */
        get() =
definedExternally
    set(value) = definedExternally
    var screenY: Int? /* = 0 */
        get() =
definedExternally
    set(value) = definedExternally
    var clientX: Int? /* = 0 */
        get() =
definedExternally
    set(value) = definedExternally
    var clientY: Int? /* = 0 */
        get() =
definedExternally
    set(value) = definedExternally
    var button: Short? /* = 0 */
        get() =
definedExternally
    set(value) = definedExternally
    var buttons: Short? /* = 0 */
        get() =
definedExternally
    set(value) = definedExternally
    var relatedTarget: EventTarget? /* = null */
        get() =
definedExternally
    set(value) = definedExternally
    var region: String? /* = null */
        get() =
definedExternally
    set(value) = definedExternally
    var detail: Int? /* = 0 */
        get() =
definedExternally
    set(value) = definedExternally
    var bubbles: Boolean? /* = false */
        get() =
definedExternally
    set(value) = definedExternally
    var cancelable: Boolean? /* = false */
        get() =
definedExternally
    set(value) = definedExternally
    var composed: Boolean? /* = false */
        get() =
definedExternally
    set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["screenX"] = screenX
    o["screenY"] = screenY
    o["clientX"] = clientX
    o["clientY"] = clientY
    o["button"] = button
    o["buttons"] = buttons
    o["relatedTarget"] = relatedTarget
    o["region"] = region
    o["ctrlKey"] = ctrlKey
    o["shiftKey"] = shiftKey
    o["altKey"] = altKey
    o["metaKey"] = metaKey
    o["modifierAltGraph"] = modifierAltGraph
    o["modifierCapsLock"] = modifierCapsLock
    o["modifierFn"] = modifierFn
    o["modifierFnLock"] = modifierFnLock
    o["modifierHyper"] = modifierHyper
    o["modifierNumLock"] = modifierNumLock
    o["modifierScrollLock"] = modifierScrollLock
    o["modifierSuper"] = modifierSuper
    o["modifierSymbol"] = modifierSymbol
    o["modifierSymbolLock"] = modifierSymbolLock
    o["view"] = view
    o["detail"] = detail
    o["bubbles"] = bubbles
    o["Cancelable"] = cancelable
    o["composed"] = composed
    return o
}

@kotlin.internal.InlineOnly
**Open Source Used In security_automation identity-guest-cs 1658**

[KeyboardEvent](https://developer.mozilla.org/en/docs/Web/API/KeyboardEvent) to Kotlin

```kotlin
open class KeyboardEvent(type: String, eventInitDict: KeyboardEventInit = definedExternally) : UIEvent {
    var key: String? /* = "" */
        get() = definedExternally
        set(value) = definedExternally
    var code: String? /* = "" */
        get() = definedExternally
        set(value) = definedExternally
    var location: Int? /* = 0 */
        get() = definedExternally
        set(value) = definedExternally
    var repeat: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var isComposing: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var ctrlKey: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var shiftKey: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var altKey: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var metaKey: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var isComposing: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var charCode: Int? /* = 0 */
        get() = definedExternally
        set(value) = definedExternally
    var keyCode: Int? /* = 0 */
        get() = definedExternally
        set(value) = definedExternally
    var which: Int? /* = 0 */
        get() = definedExternally
        set(value) = definedExternally
    fun getModifierState(keyArg: String): Boolean

    companion object {
        val DOM_KEY_LOCATION_STANDARD: Int
        val DOM_KEY_LOCATION_LEFT: Int
        val DOM_KEY_LOCATION_RIGHT: Int
        val DOM_KEY_LOCATION_NUMPAD: Int
        val NONE: Int
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public interface KeyboardEventInit : EventModifierInit {
    var key: String? /* = "" */
        get() = definedExternally
        set(value) = definedExternally
    var code: String? /* = "" */
        get() = definedExternally
        set(value) = definedExternally
    var location: Int? /* = 0 */
        get() = definedExternally
        set(value) = definedExternally
    var repeat: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var isComposing: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var ctrlKey: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var shiftKey: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var altKey: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var metaKey: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var charCode: Int? /* = 0 */
        get() = definedExternally
        set(value) = definedExternally
    var keyCode: Int? /* = 0 */
        get() = definedExternally
        set(value) = definedExternally
    var which: Int? /* = 0 */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    isComposing: Boolean? = false, ctrlKey: Boolean? = false, shiftKey: Boolean? = false, altKey: Boolean? = false,
    metaKey: Boolean? = false, modifierAltGraph: Boolean? = false, modifierCapsLock: Boolean? = false,
    modifierFn: Boolean? = false, modifierFnLock: Boolean? = false, modifierHyper: Boolean? = false,
    modifierNumLock: Boolean? = false, modifierScrollLock: Boolean? = false, modifierSuper: Boolean? = false,
    val o = js("({})")
    o["key"] = key
    o["code"] = code
    o["location"] = location
    o["repeat"] = repeat
    o["isComposing"] = isComposing
    o["ctrlKey"] = ctrlKey
    o["shiftKey"] = shiftKey
    o["altKey"] = altKey
    o["metaKey"] = metaKey
    o["modifierAltGraph"] = modifierAltGraph
    o["modifierCapsLock"] = modifierCapsLock
    o["modifierFn"] = modifierFn
    o["modifierFnLock"] = modifierFnLock
    o["modifierHyper"] = modifierHyper
    o["modifierNumLock"] = modifierNumLock
    o["modifierScrollLock"] = modifierScrollLock
    o["modifierSuper"] = modifierSuper
    o["modifierSymbol"] = modifierSymbol
    o["modifierSymbolLock"] = modifierSymbolLock
    o["view"] = view
    o["detail"] = detail
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    o["return"] = return
    o["composed"] = composed
    return o
}
```

**CompositionEvent**

```kotlin
open class CompositionEvent(type: String, eventInitDict: CompositionEventInit = definedExternally) : UIEvent {
    var data: String? /* = "" */
        get() = definedExternally
        set(value) = definedExternally
}

public interface CompositionEventInit : UIEventInit {
    var data: String? /* = "" */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun CompositionEventInit(data: String? = "", view: Window? = null, detail: Int? = 0, bubbles: Boolean? = false,
    cancelable: Boolean? = false, composed: Boolean? = false): CompositionEventInit {
    val o = js("({})")
    o["data"] = data
    o["view"] = view
    o["detail"] = detail
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}
```

**Event**

```kotlin
open class Event(type: String, eventInitDict: EventInit = definedExternally) : UIEvent {
    var target: EventTarget? /* = window */
        get() = definedExternally
    var data: Array<any>? /* = undefined */
        get() = definedExternally
}
```

[CompositionEvent](https://developer.mozilla.org/en/docs/Web/API/CompositionEvent) to Kotlin

```kotlin
@kotlin.internal.InlineOnly
public inline fun EventInit(data: Array<any>? = null, view: Window? = null, detail: Int? = 0, bubbles: Boolean? = false,
    cancelable: Boolean? = false, composed: Boolean? = false): EventInit {
    val o = js("({})")
    o["data"] = data
    o["view"] = view
    o["detail"] = detail
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}
```

[Event](https://developer.mozilla.org/en/docs/Web/API/Event) to Kotlin

```kotlin
@kotlin.internal.InlineOnly
public inline fun EventInit(data: Array<any>? = null, view: Window? = null, detail: Int? = 0, bubbles: Boolean? = false,
    cancelable: Boolean? = false, composed: Boolean? = false): EventInit {
    val o = js("({})")
    o["data"] = data
    o["view"] = view
    o["detail"] = detail
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}
```
Open Source Used In security_automation identity-guest-cs 1659
/**
 * Exposes the JavaScript [HTMLBodyElement](https://developer.mozilla.org/en/docs/Web/API/HTMLBodyElement) to Kotlin
 */
public external abstract class HTMLBodyElement : HTMLElement, WindowEventHandlers {
    open var text: String
    open var link: String
    open var vLink: String
    open var aLink: String
    open var bgColor: String
    open var background: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLHeadingElement](https://developer.mozilla.org/en/docs/Web/API/HTMLHeadingElement) to Kotlin
 */
public external abstract class HTMLHeadingElement : HTMLElement {
    open var align: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLParagraphElement](https://developer.mozilla.org/en/docs/Web/API/HTMLParagraphElement) to Kotlin
 */
public external abstract class HTMLParagraphElement : HTMLElement {
    open var align: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLHRElement](https://developer.mozilla.org/en/docs/Web/API/HTMLHRElement) to Kotlin
 */
public external abstract class HTMLHRElement : HTMLElement {
    open var align: String
    open var color: String
    open var noShade: Boolean
    open var size: String
    open var width: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
Open Source Used In security_automation identity-guest-cs 1663

Short
val PROCESSING_INSTRUCTION_NODE: Short
val COMMENT_NODE: Short
val DOCUMENT_NODE: Short
val DOCUMENT_TYPE_NODE: Short
val DOCUMENT_FRAGMENT_NODE: Short
val NOTATION_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

/**
 * Exposes the JavaScript [HTMLPreElement](https://developer.mozilla.org/en/docs/Web/API/HTMLPreElement) to Kotlin */
public external abstract class HTMLPreElement : HTMLElement {
    open var width: Int

    companion object {

val ELEMENT_NODE: Short
    val ATTRIBUTE_NODE: Short
    val TEXT_NODE: Short
    val CDATA_SECTION_NODE: Short
    val ENTITY_REFERENCE_NODE: Short
    val ENTITY_NODE: Short
    val PROCESSING_INSTRUCTION_NODE: Short
    val COMMENT_NODE: Short
    val DOCUMENT_NODE: Short
    val DOCUMENT_TYPE_NODE: Short
    val DOCUMENT_FRAGMENT_NODE: Short
    val NOTATION_NODE: Short
    val DOCUMENT_POSITION_DISCONNECTED: Short
    val DOCUMENT_POSITION_PRECEDING: Short
    val DOCUMENT_POSITION_FOLLOWING: Short
    val DOCUMENT_POSITION_CONTAINS: Short
    val DOCUMENT_POSITION_CONTAINED_BY: Short
    val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
}
DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

/**
 * Exposes the JavaScript `HTMLLIElement` to Kotlin
 */
public external abstract class HTMLLIElement : HTMLElement {
    open var value: Int
    open var type: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript `HTMLDListElement` to Kotlin
 */
public external abstract class HTMLDListElement : HTMLElement {
    open var compact: Boolean

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript `HTMLDivElement` to Kotlin
 */
public external abstract class HTMLDivElement : HTMLElement {
    open var align: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript `HTMLAnchorElement` to Kotlin
 */
public external abstract class HTMLAnchorElement : HTMLElement, HTMLHyperlinkElementUtils {
    open var target: String
    open var download: String
    open var ping: String
    open var rel: String
    open val relList: DOMTokenList
    open var hreflang: String
    open var type: String
    open var text: String
    open var referrerPolicy: String
    open var coords: String
    open var charset: String
    open var name: String
    open var rev: String
    open var shape: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
/**
 * Exposes the JavaScript [HTMLElement](https://developer.mozilla.org/en/docs/Web/API/HTMLElement) to Kotlin *
 */

public external abstract class HTMLDataElement : HTMLElement {

    open var value: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLTimeElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTimeElement) to Kotlin *
 */

public external abstract class HTMLTimeElement : HTMLElement {

    open var dateTime: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLSpanElement](https://developer.mozilla.org/en/docs/Web/API/HTMLSpanElement) to Kotlin *
 */

public external abstract class HTMLSpanElement : HTMLElement {

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLBRElement](https://developer.mozilla.org/en/docs/Web/API/HTMLBRElement) to Kotlin *
 */

public external abstract class HTMLBRElement : HTMLElement {

    open var clear: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
public interface HTMLHyperlinkElementUtils {
    var href: String
    val origin: String
    var protocol: String
    var username: String
    var password: String
    var host: String
    var hostname: String
    var port: String
    var pathname: String
    var search: String
    var hash: String
}

abstract class HTMLModElement : HTMLElement {
    open var cite: String
    open var dateTime: String
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}

abstract class HTMLPictureElement : HTMLElement {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}

abstract class HTMLSourceElement : HTMLElement {
    open var src: String
    open var type: String
    open var srcset: String
    open var sizes: String
    open var media: String
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}

abstract class HTMLImageElement : HTMLElement, HTMLOrSVGImageElement, TexImageSource {
    open var alt: String
    open var src: String
    open var srcset: String
    open var sizes: String
    open var crossOrigin: String?
    open var useMap: String
    open var isMap: Boolean
    open var width: Int
    open var height: Int
    open val naturalWidth: Int
    open val naturalHeight: Int
    open val complete: Boolean
    open var currentSrc: String
    open var referrerPolicy: String
    open var name: String
    open var lowsrc: String
    open var align: String
    open var hspace: Int
    open var vspace: Int
    open var longDesc: String
    open var border: String
    open var x: Int
    open var y: Int
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}
CDATA_SECTION_NODE: Short
val ENTITY_REFERENCE_NODE: Short
val ENTITY_NODE: Short
val PROCESSING_INSTRUCTION_NODE: Short
val COMMENT_NODE: Short
val DOCUMENT_NODE: Short
val DOCUMENT_TYPE_NODE: Short
val DOCUMENT_FRAGMENT_NODE: Short
val NOTATION_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
}

text
}

/**
 * Exposes the JavaScript
[HTMLIFrameElement](https://developer.mozilla.org/en/docs/Web/API/HTMLIFrameElement) to Kotlin
*/
public external abstract class HTMLIFrameElement : HTMLElement {
    open var src: String
    open var srcdoc: String
    open var name: String
    open val sandbox: DOMTokenList
    open var allowFullscreen: Boolean
    open var allowUserMedia: Boolean
    open var width: String
    open var height: String
    open var referrerPolicy: String
    open var contentDocument: Document?
    open var contentWindow: Window?
    open var align: String
    open var scrolling: String
    open var frameborder: String
    open var longDesc: String
    open var marginHeight: String
    open var marginWidth: String
    fun getSVGDocument(): Document?

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }

/**
 * Exposes the JavaScript
[HTMLEmbedElement](https://developer.mozilla.org/en/docs/Web/API/HTMLEmbedElement) to Kotlin
*/
public external abstract class HTMLEmbedElement : HTMLElement {
    open var src: String
    open var type: String
    open var width: String
    open var height: String
    open var align: String
    open var name: String
    fun getSVGDocument(): Document?

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }

/**
 * Exposes the JavaScript
[HTMLObjectElement](https://developer.mozilla.org/en/docs/Web/API/HTMLObjectElement) to Kotlin
*/
public external abstract class HTMLObjectElement : HTMLElement {
    open var data: String
    open var type: String
    open var typeMustMatch: Boolean
    open var name: String
    open var useMap: String
    open val form: HTMLFormElement?
    open var width: String
    open var height: String
    open val contentDocument: Document?
    open val contentWindow: Window?
    open var willValidate: Boolean
    open var validity: ValidityState
    open var validationMessage: String
    open var align: String
    open var archive: String
    open var code: String
    open var declare: Boolean
    open var hspace: Int
    open var standby: String
    open var vspace: Int
    open var codeBase: String
    open var codeType: String
    open var border: String
    fun getSVGDocument(): Document?
    fun checkValidity(): Boolean
    fun reportValidity(): Boolean
    fun setCustomValidity(error: String)

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
    }

Open Source Used In security_automation identity-guest-cs 1667
val ENTITY_REFERENCE_NODE: Short
val ENTITY_NODE: Short
val PROCESSING_INSTRUCTION_NODE: Short
val COMMENT_NODE: Short
val DOCUMENT_NODE: Short
val DOCUMENT_TYPE_NODE: Short
val DOCUMENT_FRAGMENT_NODE: Short
val NOTATION_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

/**
 * Exposes the JavaScript [HTMLParamElement](https://developer.mozilla.org/en/docs/Web/API/HTMLParamElement) to Kotlin
 */
public external abstract class HTMLParamElement : HTMLElement {
    open var name: String
    open var value: String
    open var type: String
    open var valueType: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLVideoElement](https://developer.mozilla.org/en/docs/Web/API/HTMLVideoElement) to Kotlin
 */
public external abstract class HTMLVideoElement : HTMLMediaElement, CanvasImageSource, TexImageSource {
    open var width: Int
    open var height: Int
    open val videoWidth: Int
    open val videoHeight: Int
    open var poster: String
    open var playsInline: Boolean

    companion object {
        val NETWORK_EMPTY: Short
        val NETWORK_IDLE: Short
        val NETWORK_LOADING: Short
        val NETWORK_NO_SOURCE: Short
        val HAVE_NOTHING: Short
        val HAVE_METADATA: Short
        val HAVE_CURRENT_DATA: Short
        val HAVE_FUTURE_DATA: Short
        val HAVE_ENOUGH_DATA: Short
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}

/**
 * Exposes the JavaScript [HTMLAudioElement](https://developer.mozilla.org/en/docs/Web/API/HTMLAudioElement) to Kotlin
 */
public external abstract class HTMLAudioElement : HTMLMediaElement {
    companion object {
        val NETWORK_EMPTY: Short
        val NETWORK_IDLE: Short
        val NETWORK_LOADING: Short
        val NETWORK_NO_SOURCE: Short
        val HAVE_NOTHING: Short
        val HAVE_METADATA: Short
        val HAVE_CURRENT_DATA: Short
        val HAVE_FUTURE_DATA: Short
        val HAVE_ENOUGH_DATA: Short
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

// Exposes the JavaScript [HTMLTrackElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTrackElement) to Kotlin

```kotlin
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION(dirname)
abstract class AudioTrackList : EventTarget {
    open val length: Int
    open var onaddtrack: ((TrackEvent) -> dynamic)?
    open var onremovetrack: ((TrackEvent) -> dynamic)?
    fun getTrackById(id: String): AudioTrack?
}

@kotlin.internal.InlineOnly
public inline operator fun AudioTrackList.get(index: Int): AudioTrack? = asDynamic()[index]

/**
 * Exposes the JavaScript [AudioTrack](https://developer.mozilla.org/en/docs/Web/API/AudioTrack) to Kotlin
 */
public external abstract class AudioTrack : EventTarget, UnionAudioTrackOrTextTrackOrVideoTrack {
    open val id: String
    open val kind: String
    open val label: String
    open val language: String
    open var enabled: Boolean
    open val sourceBuffer: SourceBuffer?
}

/**
 * Exposes the JavaScript [VideoTrackList](https://developer.mozilla.org/en/docs/Web/API/VideoTrackList) to Kotlin
 */
public external abstract class VideoTrackList : EventTarget {
    open val length: Int
    open val selectedIndex: Int
    open var onchange: ((Event) -> dynamic)?
    open var onaddtrack: ((TrackEvent) -> dynamic)?
    open var onremovetrack: ((TrackEvent) -> dynamic)?
    fun getTrackById(id: String): VideoTrack?
}

@kotlin.internal.InlineOnly
public inline operator fun VideoTrackList.get(index: Int): VideoTrack? = asDynamic()[index]

/**
 * Exposes the JavaScript [VideoTrack](https://developer.mozilla.org/en/docs/Web/API/VideoTrack) to Kotlin
 */
public external abstract class VideoTrack : EventTarget, UnionAudioTrackOrTextTrackOrVideoTrack {
    open val id: String
    open val kind: String
    open val label: String
    open val language: String
    open var selected: Boolean
    open val sourceBuffer: SourceBuffer?
}

/**
 * Exposes the JavaScript [TextTrackList](https://developer.mozilla.org/en/docs/Web/API/TextTrackList) to Kotlin
 */
public external abstract class TextTrackList : EventTarget {
    open val length: Int
    open var onchange: ((Event) -> dynamic)?
    open var onaddtrack: ((TrackEvent) -> dynamic)?
    open var onremovetrack: ((TrackEvent) -> dynamic)?
    fun getTrackById(id: String): TextTrack?
}

@kotlin.internal.InlineOnly
public inline operator fun TextTrackList.get(index: Int): TextTrack? = asDynamic()[index]

/**
 * Exposes the JavaScript [TextTrack](https://developer.mozilla.org/en/docs/Web/API/TextTrack) to Kotlin
 */
public external abstract class TextTrack : EventTarget, UnionAudioTrackOrTextTrackOrVideoTrack {
    open val kind: TextTrackKind
    open val label: String
    open val language: String
    open val id: String
    open val inBandMetadataTrackDispatchType: String
    open var mode: TextTrackMode
    open val cues: TextTrackCueList?
    open val activeCues: TextTrackCueList?
    open var oncuechange: ((Event) -> dynamic)?
    open val sourceBuffer: SourceBuffer?
    fun addCue(cue: TextTrackCue)
    fun removeCue(cue: TextTrackCue)
}

public external abstract class TextTrackCueList {
    open val length: Int
    fun getCueById(id: String): TextTrackCue?
}

@kotlin.internal.InlineOnly
public inline operator fun TextTrackCueList.get(index: Int): TextTrackCue? = asDynamic()[index]

/**
 * Exposes the JavaScript [TextTrackCue](https://developer.mozilla.org/en/docs/Web/API/TextTrackCue) to Kotlin
 */
public external abstract class TextTrackCue : EventTarget {
    open var track: TextTrack?
    open var id: String
    open var startTime: Double
    open var endTime: Double
    open var pauseOnExit: Boolean
    open var onenter: ((Event) -> dynamic)?
    open var onexit: ((Event) -> dynamic)?
}

/**
 * Exposes the JavaScript [TimeRanges](https://developer.mozilla.org/en/docs/Web/API/TimeRanges) to Kotlin
 */
public external abstract class TimeRanges {
    open val length: Int
    fun start(index: Int): Double
    fun end(index: Int): Double
}

/**
 * Exposes the JavaScript [TrackEvent](https://developer.mozilla.org/en/docs/Web/API/TrackEvent) to Kotlin
 */
public external open class TrackEvent(type: String, eventInitDict: TrackEventInit = definedExternally) : Event {
    open val track: UnionAudioTrackOrTextTrackOrVideoTrack?

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public external interface TrackEventInit : EventInit {
    var track: UnionAudioTrackOrTextTrackOrVideoTrack? /* = null */
        get() = definedExternally
        set(value) = definedExternally
    }

@kotlin.internal.InlineOnly
    return definedExternally
}

Open Source Used In security Automation Identity-Guest-cs 1670
JavaScript [HTMLMapElement](https://developer.mozilla.org/en/docs/Web/API/HTMLMapElement) to Kotlin

```kotlin
/**
 * Exposes the JavaScript [HTMLAreaElement](https://developer.mozilla.org/en/docs/Web/API/HTMLAreaElement) to Kotlin
 */
public external abstract class HTMLAreaElement : HTMLElement, HTMLHyperlinkElementUtils {
    open var alt: String
    open var coords: String
    open var shape: String
    open var target: String
    open var download: String
    open var ping: String
    open var rel: String
    open val relList: DOMTokenList
    open var referrerPolicy: String
    open var noHref: Boolean

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
```

```kotlin
/**
 * Exposes the JavaScript [HTMLTableElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTableElement) to Kotlin
 */
public external abstract class HTMLTableElement : HTMLElement {
    open var caption: HTMLTableCaptionElement?
    open var tHead: HTMLTableSectionElement?
    open var tFoot: HTMLTableSectionElement?
    open val tBodies: HTMLCollection
    open val rows: HTMLCollection
    open var align: String
    open var border: String
    open var frame: String
    open var rules: String
    open var summary: String
    open var width: String
    open var bgColor: String
    open var cellPadding: String
    open var cellSpacing: String
    fun createCaption(): HTMLTableCaptionElement
    fun deleteCaption()
    fun createTHead(): HTMLTableSectionElement
    fun deleteTHead()
    fun createTFoot(): HTMLTableSectionElement
    fun deleteTFoot()
    fun insertRow(index: Int = definedExternally): HTMLTableRowElement
    fun deleteRow(index: Int)

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
```

```kotlin
/**
 * Exposes the JavaScript [HTMLTableCaptionElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTableCaptionElement) to Kotlin
 */
public external abstract class HTMLTableCaptionElement : HTMLElement {
    open var align: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
    }
}
```
*/
public external abstract class HTMLTableCellElement : HTMLElement {
    open var colSpan: Int
    open var rowSpan: Int
    open var align: String
    open var accessKey: String
    open var accessKey: String
    open var axis: String
    open var border: Int
    open var borderCollapse: String
    open var borderSpacing: String
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<Element>
    open var children: List<El...
open val willValidate: Boolean
open val validity: ValidityState
open val validationMessage: String
open val labels: NodeList
fun namedItem(name: String): HTMLOptionElement
fun add(element: UnionHTMLOptGroupElementOrHTMLOptionElement, before: dynamic = definedExternally)
fun remove(index: Int)
fun checkValidity(): Boolean
fun reportValidity(): Boolean
fun setCustomValidity(error: String)
override fun item(index: Int): Element?

companion object {
val ELEMENT_NODE: Short
val ATTRIBUTE_NODE: Short
val TEXT_NODE: Short
val CDATA_SECTION_NODE: Short
val ENTITY_REFERENCE_NODE: Short
val ENTITY_NODE: Short
val PROCESSING_INSTRUCTION_NODE: Short
val COMMENT_NODE: Short
val DOCUMENT_NODE: Short
val DOCUMENT_TYPE_NODE: Short
val DOCUMENT_FRAGMENT_NODE: Short
val NOTATION_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
}

@kotlin.internal.InlineOnly
public inline operator fun HTMLSelectElement.get(index: Int): Element? = asDynamic()[index]

@kotlin.internal.InlineOnly
public inline operator fun HTMLSelectElement.set(index: Int, option: HTMLOptionElement?) { asDynamic()[index] = option }

/**
 * Exposes the JavaScript [HTMLDataListElement](https://developer.mozilla.org/en/docs/Web/API/HTMLDataListElement) to Kotlin
 */
public external abstract class HTMLDataListElement : HTMLElement {
open val options: HTMLCollection

companion object {
val ELEMENT_NODE: Short
val ATTRIBUTE_NODE: Short
val TEXT_NODE: Short
val CDATA_SECTION_NODE: Short
val ENTITY_REFERENCE_NODE: Short
val ENTITY_NODE: Short
val PROCESSING_INSTRUCTION_NODE: Short
val COMMENT_NODE: Short
val DOCUMENT_NODE: Short
val DOCUMENT_TYPE_NODE: Short
val DOCUMENT_FRAGMENT_NODE: Short
val NOTATION_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
}

/**
 * Exposes the JavaScript [HTMLOptGroupElement](https://developer.mozilla.org/en/docs/Web/API/HTMLOptGroupElement) to Kotlin
 */
public external abstract class HTMLOptGroupElement : HTMLElement, UnionHTMLOptGroupElementOrHTMLOptionElement {
open var disabled: Boolean
open var label: String

companion object {
val ELEMENT_NODE: Short
val ATTRIBUTE_NODE: Short
val TEXT_NODE: Short
val CDATA_SECTION_NODE: Short
val ENTITY_REFERENCE_NODE: Short
val ENTITY_NODE: Short
val PROCESSING_INSTRUCTION_NODE: Short
val COMMENT_NODE: Short
val DOCUMENT_NODE: Short
val DOCUMENT_TYPE_NODE: Short
val DOCUMENT_FRAGMENT_NODE: Short
val NOTATION_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
}

/**
 * Exposes the JavaScript [HTMLOptionElement](https://developer.mozilla.org/en/docs/Web/API/HTMLOptionElement) to Kotlin
 */
public external abstract class HTMLOptionElement : HTMLElement, UnionHTMLOptGroupElementOrHTMLOptionElement {
open var disabled: Boolean
open var form: HTMLFormElement?
open var label: String
open var defaultSelected: Boolean
open var selected: Boolean
open var value: String
open var text: String
open var index: Int

companion object {
val ELEMENT_NODE: Short
val ATTRIBUTE_NODE: Short
val TEXT_NODE: Short
val CDATA_SECTION_NODE: Short
val ENTITY_REFERENCE_NODE: Short
val ENTITY_NODE: Short
val PROCESSING_INSTRUCTION_NODE: Short
val COMMENT_NODE: Short
val DOCUMENT_NODE: Short
val DOCUMENT_TYPE_NODE: Short
val DOCUMENT_FRAGMENT_NODE: Short
val NOTATION_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
}
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short

DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

\*/

* Exposes the JavaScript [HTMLLegendElement](https://developer.mozilla.org/en/docs/Web/API/HTMLLegendElement) to Kotlin

\*/

public external abstract class HTMLLegendElement : HTMLElement {
    open val form: HTMLFormElement?
    open var align: String?

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [ValidityState](https://developer.mozilla.org/en/docs/Web/API/ValidityState) to Kotlin
 */
public external abstract class ValidityState {
    open val valueMissing: Boolean
    open val typeMismatch: Boolean
    open val patternMismatch: Boolean
    open val tooLong: Boolean
    open val tooShort: Boolean
    open val rangeUnderflow: Boolean
    open val rangeOverflow: Boolean
    open val badInput: Boolean
    open val customError: Boolean
    open val valid: Boolean
}

/**
 * Exposes the JavaScript [HTMLDetailsElement](https://developer.mozilla.org/en/docs/Web/API/HTMLDetailsElement) to Kotlin
 */
public external abstract class HTMLDetailsElement : HTMLElement {
    open var open: Boolean

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

public external abstract class HTMLMenuElement : HTMLElement {
    open var type: String
    open var label: String
    open var compact: Boolean

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

public external abstract class HTMLMenuItemElement : HTMLElement {
    open var type: String
    open var label: String
    open var icon: String
    open var disabled: Boolean
    open var checked: Boolean
    open var radiogroup: String
    open var default: Boolean

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
/**
 * Exposes the JavaScript [HTMLSlotElement](https://developer.mozilla.org/en/docs/Web/API/HTMLSlotElement) to Kotlin
 */
public external abstract class HTMLSlotElement : HTMLElement {
    open var name: String
    fun assignedNodes(options: AssignedNodesOptions = definedExternally): Array<Node>

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

public external interface AssignedNodesOptions {
    var flatten: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun AssignedNodesOptions(flatten: Boolean? = false): AssignedNodesOptions {
    val o = js("({})")
    o["flatten"] = flatten
    return o
}

/**
 * Exposes the JavaScript [HTMLCanvasElement](https://developer.mozilla.org/en/docs/Web/API/HTMLCanvasElement) to Kotlin
 */
public external abstract class HTMLCanvasElement : HTMLElement, CanvasImageSource, TexImageSource {
    open var width: Int
    open var height: Int
    fun getContext(contextId: String, vararg arguments: Any?): RenderingContext?
    fun toDataURL(type: String = definedExternally, quality: Any? = definedExternally): String
    fun toBlob(_callback: (Blob?) -> Unit, type: String = definedExternally, quality: Any? = definedExternally)

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

public external interface CanvasRenderingContext2DSettings {
    var alpha: Boolean? /* = true */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun CanvasRenderingContext2DSettings(alpha: Boolean? = true): CanvasRenderingContext2DSettings {
    val o = js("({})")
    o["alpha"] = alpha
    return o
}

/**
 * Exposes the JavaScript [CanvasRenderingContext2D](https://developer.mozilla.org/en/docs/Web/API/CanvasRenderingContext2D) to Kotlin
 */
public external abstract class CanvasRenderingContext2D : CanvasState, CanvasTransform, CanvasCompositing, CanvasImageSmoothing, CanvasFillStrokeStyles, CanvasShadowStyles, CanvasFilters, CanvasRect, CanvasDrawPath, CanvasUserInterface, CanvasText, CanvasDrawImage, CanvasHitRegion, CanvasImageData, CanvasPathDrawingStyles, CanvasTextDrawingStyles, CanvasPath, RenderingContext {
    open val canvas: HTMLCanvasElement
}

public external interface CanvasState {
    fun save()
    fun restore()
}

public external interface CanvasTransform {
    fun scale(x: Double, y: Double)
    fun rotate(angle: Double)
    fun translate(x: Double, y: Double)
    fun transform(a: Double, b: Double, c: Double, d: Double, e: Double, f: Double)
    fun getTransform(): DOMMatrix
}
abstract class CanvasGradient {
    fun addColorStop(offset: Double, color: String)
}

public abstract class CanvasGradient {
    fun addColorStop(offset: Double, color: String)
}

abstract class CanvasGradient {
    fun addColorStop(offset: Double, color: String)
}

/**
 * Exposes the JavaScript [CanvasPattern](https://developer.mozilla.org/en/docs/Web/API/CanvasPattern) to Kotlin
 */
public abstract class CanvasGradient {
    fun addColorStop(offset: Double, color: String)
}

/**
 * Exposes the JavaScript [CanvasPattern](https://developer.mozilla.org/en/docs/Web/API/CanvasPattern) to Kotlin
 */
public abstract class CanvasGradient {
    fun addColorStop(offset: Double, color: String)
}

/**
 * Exposes the JavaScript [CanvasPattern](https://developer.mozilla.org/en/docs/Web/API/CanvasPattern) to Kotlin
 */
public abstract class CanvasGradient {
    fun addColorStop(offset: Double, color: String)
}

/**
 * Exposes the JavaScript [CanvasPattern](https://developer.mozilla.org/en/docs/Web/API/CanvasPattern) to Kotlin
 */
public abstract class CanvasGradient {
    fun addColorStop(offset: Double, color: String)
}

/**
 * Exposes the JavaScript [CanvasPattern](https://developer.mozilla.org/en/docs/Web/API/CanvasPattern) to Kotlin
 */
public abstract class CanvasGradient {
    fun addColorStop(offset: Double, color: String)
}

/**
 * Exposes the JavaScript [CanvasPattern](https://developer.mozilla.org/en/docs/Web/API/CanvasPattern) to Kotlin
 */
public abstract class CanvasGradient {
    fun addColorStop(offset: Double, color: String)
}
options: ElementDefinitionOptions = definedExternally\n    fun get(name: String): Any?\n    fun whenDefined(name: String): Promise<Unit>\n
public external interface ElementDefinitionOptions {
    val o = js(`{}`)\n    o["extends"] = extends\n    return o\n}

public external interface ElementContentEditable {
    var contentEditable: Boolean?\n    isContentEditable: Boolean\n}

@kotlin.internal.InlineOnly
public inline fun ElementDefinitionOptions(extends: String? = undefined): ElementDefinitionOptions {
    val o = js(`{}`)\n    o["extends"] = extends\n    return o\n}

public external fun ElementContentEditable(contentEditable: Boolean? = undefined): ElementContentEditable\n
[DataTransfer](https://developer.mozilla.org/en/docs/Web/API/DataTransfer) to Kotlin\n
**Exposes the JavaScript abstract class DataTransfer\n
open var dropEffect: String?\n
fun getAsFile(): File?\n
setDragImage(image: Element, x: Int, y: Int)\n
fun getData(format: String): String?\n
fun setData(format: String, data: String?)\n
fun clearData(format: String = definedExternally)\n
[DataTransferItemList](https://developer.mozilla.org/en/docs/Web/API/DataTransferItemList) to Kotlin\n
**Exposes the JavaScript abstract class DataTransferItemList\n
open val length: Int\n
fun add(data: String, type: String): DataTransferItem?\n
fun add(data: File): DataTransferItem?\n
fun remove(index: Int)\n
fun clear()\n
[DataTransferItem](https://developer.mozilla.org/en/docs/Web/API/DataTransferItem) to Kotlin\n
**Exposes the JavaScript abstract class DataTransferItem\n
open val kind: String\n
fun getAsString(callback: ((String) -> Unit)?)\n
fun getAsFile(): File?\n
[DragEvent](https://developer.mozilla.org/en/docs/Web/API/DragEvent) to Kotlin\n
**Exposes the JavaScript \n
DragEvent(type: string, eventInitDict: DragEventInit = definedExternally) : MouseEvent\n
DragEventInit : DragEventInit\n
[Window](https://developer.mozilla.org/en/docs/Web/API/Window) to Kotlin\n
**Exposes the JavaScript abstract class Window\n
open val window: Window\n
open var dropEffect: String?\n
fun setDragImage(image: Element, x: Int, y: Int)\n
fun getData(format: String): String?\n
fun setData(format: String, data: String?)\n
fun clearData(format: String = definedExternally)\n
open val screenX: Int?\n
open val screenY: Int?\n
open val clientX: Int?\n
open val clientY: Int?\n
open val button: Short?\n
open val buttons: Short?\n
open val relatedTarget: EventTarget?\n
open val region: String?\n
open val ctrlKey: Boolean?\n
open val shiftKey: Boolean?\n
open val altKey: Boolean?\n
open val metaKey: Boolean?\n
open val modifierAltGraph: Boolean?\n
open val modifierCapsLock: Boolean?\n
open val modifierFn: Boolean?\n
open val modifierFnLock: Boolean?\n
open val modifierHyper: Boolean?\n
open val modifierNumLock: Boolean?\n
open val modifierScrollLock: Boolean?\n
open val modifierSuper: Boolean?\n
open val modifierSymbol: Boolean?\n
open val modifierSymbolLock: Boolean?\n
open val view: Window?\n
open val detail: Int?\n
open val bubbles: Boolean?\n
open val cancelable: Boolean?\n
open val composed: Boolean?\n
set(value: definedExternally)\n
fun ElementContentEditable(contentEditable: Boolean? = undefined): ElementContentEditable\n
fun whenDefined(name: String): Promise<Unit>
open val customElements: CustomElementRegistry
    open val locationbar: BarProp
    open val menubar: BarProp
    open val personalbar: BarProp
    open val scrollbars: BarProp
    open val statusbar: BarProp
    open val toolbar: BarProp
    open var status: String
    open val closed: Boolean
    open val frames: Window
    open val length: Int
    open val top: Window
    open var opener: Any?
    open val parent: Window
    open val frameElement: Element?
    open val navigator: Navigator
    open val applicationCache: ApplicationCache
    open val external: External
    open val open screen: Screen
    open val innerWidth: Int
    open val innerHeight: Int
    open val scrollX: Double
    open val pageXOffset: Double
    open val scrollY: Double
    open val pageYOffset: Double
    open val open screenX: Int
    open val open screenY: Int
    open val open outerHeight: Int
    open val open devicePixelRatio: Double
    fun close()
    fun stop()
    fun focus()
    fun blur()
    fun open(url: String = definedExternally, target: String = definedExternally, features: String = definedExternally): Window?
    fun alert()
    fun alert(message: String)
    fun confirm(message: String = definedExternally): Boolean
    fun prompt(message: String = definedExternally, default: String = definedExternally): String?
    fun print()
    fun requestAnimationFrame(callback: (Double) -> Unit): Int
    fun cancelAnimationFrame(handle: Int)
    fun captureEvents()
    fun releaseEvents()
    fun matchMedia(query: String): MediaQueryList
    fun moveTo(x: Int, y: Int)
    fun moveBy(x: Int, y: Int)
    fun resizeTo(x: Int, y: Int)
    fun resizeBy(x: Int, y: Int)
    fun scroll(options: ScrollToOptions = definedExternally)
    fun scroll(x: Double, y: Double)
    fun scroll(x: Double, y: Double)
    fun scrollBy(options: ScrollToOptions = definedExternally)
    fun scrollBy(x: Double, y: Double)
    fun scrollBy(x: Double, y: Double)
    fun getComputedStyle(elt: Element, pseudoElt: String? = definedExternally): CSSStyleDeclaration
}

@kotlin.internal.InlineOnly
public inline operator fun Window.get(name: String): dynamic = asDynamic()[name]

public external abstract class BarProp {
    open val visible: Boolean
}

/**
 * Exposes the JavaScript
 * [History](https://developer.mozilla.org/en/docs/Web/API/History) to Kotlin
 */
public external abstract class History {
    open val length: Int
    open var scrollRestoration: ScrollRestoration
    open val state: Any?
    fun go(delta: Int = definedExternally)
    fun back()
    fun forward()
    fun pushState(data: Any?, title: String, url: String? = definedExternally)
    fun replaceState(data: Any?, title: String, url: String? = definedExternally)
}

/**
 * Exposes the JavaScript
 * [Location](https://developer.mozilla.org/en/docs/Web/API/Location) to Kotlin
 */
public external open class Location {
    open var href: String
    open val origin: String
    open var protocol: String
    open var host: String
    open var hostname: String
    open var port: String
    open var pathname: String
    open var search: String
    open var hash: String
    open val ancestorOrigins: Array<out String>
    fun assign(url: String)
    fun replace(url: String)
    fun reload()
}

/**
 * Exposes the JavaScript
 * [PopStateEvent](https://developer.mozilla.org/en/docs/Web/API/PopStateEvent) to Kotlin
 */
public external open class PopStateEvent(type: String, eventInitDict: PopStateEventInit = definedExternally) : Event {
    open val state: Any?
    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public external interface PopStateEventInit : EventInit {
    var state: Any? /* = null */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["state"] = state
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    o["composed"] = composed
    o["composed"] = composed
    o["composed"] = composed
}

/**
 * Exposes the JavaScript
 * [HashChangeEvent](https://developer.mozilla.org/en/docs/Web/API/HashChangeEvent) to Kotlin
 */
public external open class HashChangeEvent(type: String, eventInitDict: HashChangeEventInit = definedExternally) : Event {
    open val oldURL: String
    open val newURL: String
    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
    }
}

public external interface HashChangeEventInit : EventInit {
    var oldURL: String? /* = "" */
        get() = definedExternally
        set(value) = definedExternally
    var newURL: String? /* = "" */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun HashChangeEventInit(oldURL: String? = null, newURL: String? = null): HashChangeEventInit {
    val o = js("({})")
    o["oldURL"] = oldURL
    o["newURL"] = newURL
    return o
}

Open Source Used In security_automation identity-guest-cs 1684
Any?
get() = definedExternally
set(value) =
definedExternally

@kotlin.internal.InlineOnly
val o = js("({})")
o["promise"] = promise
o["reason"] = reason
o["bubbles"] = bubbles
o["Cancelable"] = cancelable
o["composed"] = composed
return o
}

/**
 * Exposes the JavaScript
 * [GlobalEventHandlers](https://developer.mozilla.org/en/docs/Web/API/GlobalEventHandlers) to Kotlin
 */
public external interface GlobalEventHandlers {
    var onabort: (Event) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onblur: (FocusEvent) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var oncancel: (Event) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var oncanplay: (Event) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var oncanplaythrough: (Event) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onchange: (Event) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onclick: (MouseEvent) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onclose: (Event) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var oncontextmenu: (MouseEvent) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var oncuechange: (Event) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var ondblclick: (MouseEvent) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var ondrag: (DragEvent) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var ondragend: (DragEvent) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var ondragenter: (DragEvent) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var ondragexit: (DragEvent) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var ondragleave: (DragEvent) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var ondragover: (DragEvent) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var ondragstart: (DragEvent) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var ondrop: (DragEvent) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var ondurationchange: (Event) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onenompted: (Event) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onerror: (dynamic, String, Int, Int, Any?) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onfocus: (FocusEvent) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var oninput: (InputEvent) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var oninvalid: (Event) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onkeydown: (KeyboardEvent) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onkeypress: (KeyboardEvent) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onkeyup: (KeyboardEvent) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onmessage: (MessageEvent) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onmousedown: (MouseEvent) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onmouseenter: (MouseEvent) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onmouseleave: (MouseEvent) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onmousemove: (MouseEvent) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onmouseout: (MouseEvent) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onmouseover: (MouseEvent) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onmousemove: (MouseEvent) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onmouseout: (MouseEvent) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onmousemove: (MouseEvent) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onmouseout: (MouseEvent) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onmousemove: (MouseEvent) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onmouseout: (MouseEvent) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onmousemove: (MouseEvent) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onmouseout: (MouseEvent) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onmousemove: (MouseEvent) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onmouseout: (MouseEvent) -> dynamic?
var onmouseover: ((MouseEvent) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
var onmousemove: ((MouseEvent) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
var onmouseout: ((MouseEvent) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
var onmousedown: ((MouseEvent) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
var onmouseup: ((MouseEvent) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
var onwheel: ((WheelEvent) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
var onpause: ((Event) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
var onplay: ((Event) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
var onplaying: ((Event) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
var onprogress: ((ProgressEvent) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
var onreset: ((Event) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
var onresize: ((Event) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
var onscroll: ((Event) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
var onseeked: ((Event) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
var onseeking: ((Event) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
var onselect: ((Event) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
var onshow: ((Event) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
var onstalled: ((Event) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
var onsubmit: ((Event) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
var onsuspend: ((Event) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
var ontimeupdate: ((Event) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
var ontoggle: ((Event) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
var onvolumechange: ((Event) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
var onwaiting: ((Event) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
var ongotpointercapture: ((PointerEvent) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
var onlostpointercapture: ((PointerEvent) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
var onpointerdown: ((PointerEvent) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
var onpointermove: ((PointerEvent) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
var onpointerup: ((PointerEvent) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
var onpointercancel: ((PointerEvent) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
var onpointerenter: ((PointerEvent) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
var onpointerleave: ((PointerEvent) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
var onbeforeprint: ((Event) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
var onbeforeunload: ((BeforeUnloadEvent) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
var onhashchange: ((HashChangeEvent) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
var onlanguagechange: ((Event) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
var onmessage: ((MessageEvent) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
var onoffline: ((Event) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
var ononline: ((Event) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
var onpagehide: ((PageTransitionEvent) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
var onpageshow: ((PageTransitionEvent) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
var onrejectionhandled: ((Event) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally

get() = definedExternally; set(value) = definedExternally; var onstorage: ((StorageEvent) -> dynamic)?
get() = definedExternally; set(value) = definedExternally; var onunhandledrejection:
((PromiseRejectionEvent) -> dynamic)?
get() = definedExternally; set(value) = definedExternally; var onunload: ((Event) -> dynamic).Bus
get() = definedExternally; set(value) = definedExternally; var oncopy:
((ClipboardEvent) -> dynamic)?
get() = definedExternally; set(value) = definedExternally; var oncut:
((ClipboardEvent) -> dynamic)?
get() = definedExternally; set(value) = definedExternally; var onpaste: ((ClipboardEvent) -> dynamic)?
get() = definedExternally; set(value) = definedExternally;}

public external interface DocumentAndElementEventHandlers {
    var oncopy: ((ClipboardEvent) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
    var oncut: ((ClipboardEvent) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
    var onpaste: ((ClipboardEvent) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
}

/**
 * Exposes the JavaScript
 */
public external interface WindowOrWorkerGlobalScope {
    val origin: String
    val caches: CacheStorage
    fun btoa(data: String): String
    fun atob(data: String): String
    fun setTimeout(handler: dynamic, timeout: Int = definedExternally, vararg arguments: Any?): Int
    fun clearTimeout(handle: Int = definedExternally)
    fun setInterval(handler: dynamic, timeout: Int = definedExternally, vararg arguments: Any?): Int
    fun clearInterval(handle: Int = definedExternally)
    fun createImageBitmap(image: ImageBitmapSource, options: ImageBitmapOptions = definedExternally): Promise<ImageBitmap>
    fun fetch(input: dynamic, init: RequestInit = definedExternally): Promise<Response>
}

/**
 * Exposes the JavaScript
 */
public external abstract class Navigator : NavigatorID, NavigatorLanguage, NavigatorOnLine, NavigatorContentUtils, NavigatorCookies, NavigatorPlugins, NavigatorConcurrentHardware {
    open val clipboard: Clipboard
    open val mediaDevices: MediaDevices
    open val maxTouchPoints: Int
    open val serviceWorker: ServiceWorkerContainer
    fun requestMediaKeySystemAccess(keySystem: String, supportedConfigurations: Array<MediaKeySystemConfiguration>): Promise<MediaKeySystemAccess>
    fun getUserMedia(constraints: MediaStreamConstraints, successCallback: (MediaStream) -> Unit, errorCallback: (dynamic) -> Unit)
    fun vibrate(pattern: dynamic): Boolean
}

/**
 * Exposes the JavaScript
 */
public external interface NavigatorID {
    val appCodeName: String
    val appName: String
    val appVersion: String
    val platform: String
    val product: String
    val productSub: String
    val userAgent: String
    val vendor: String
    val vendorSub: String
    val osCPU: String
    fun taintEnabled(): Boolean
}

/**
 * Exposes the JavaScript
 */
public external interface NavigatorLanguage {
    val language: String
    val languages: Array<out String>
}

public external interface NavigatorContentUtils {
    fun registerProtocolHandler(scheme: String, url: String, title: String)
    fun registerContentHandler(mimeType: String, url: String, title: String)
    fun isProtocolHandlerRegistered(scheme: String, url: String): String
    fun isContentHandlerRegistered(mimeType: String, url: String): String
    fun unregisterProtocolHandler(scheme: String, url: String)
    fun unregisterContentHandler(mimeType: String, url: String)
    fun unregisterContentHandlerRegistered(scheme: String, url: String)
}

public external interface NavigatorCookies {
    val cookieEnabled: Boolean
}

/**
 * Exposes the JavaScript
 */
public external interface NavigatorPlugins {
    val plugins: PluginArray
    val mimeTypes: MimeTypeArray
    fun javaEnabled(): Boolean
}

@kotlin.internal.InlineOnly
public inline operator fun PluginArray.get(index: Int): Plugin? = asDynamic()[(index)@kotlin.internal.InlineOnly]
Open Source Used In security_automation identity-guest-cs 1689

[MimeTypeArray](https://developer.mozilla.org/en/docs/Web/API/MimeTypeArray) to Kotlin

```kotlin
public external abstract class MimeTypeArray : ItemArrayLike<MimeType> {
    override fun item(index: Int): MimeType?

    fun namedItem(name: String): MimeType?
}

@kotlin.internal.InlineOnly
public inline operator fun MimeTypeArray.get(index: Int): MimeType? = asDynamic()[index]

@kotlin.internal.InlineOnly
public inline operator fun MimeTypeArray.get(name: String): MimeType? = asDynamic()[name]
```

Exposes the JavaScript [Plugin](https://developer.mozilla.org/en/docs/Web/API/Plugin) to Kotlin

```kotlin
public external abstract class Plugin : ItemArrayLike<MimeType> {
    open val name: String
    open val description: String
    open val filename: String
    override fun item(index: Int): MimeType?

    fun namedItem(name: String): MimeType?
}
```

Exposes the JavaScript [MimeType](https://developer.mozilla.org/en/docs/Web/API/MimeType) to Kotlin

```kotlin
public external abstract class MimeType {
    open val type: String
    open val description: String
    open val suffixes: String
    open val enabledPlugin: Plugin
}
```

Exposes the JavaScript [ImageBitmap](https://developer.mozilla.org/en/docs/Web/API/ImageBitmap) to Kotlin

```kotlin
public external abstract class ImageBitmap : CanvasImageSource, TexImageSource {
    open val width: Int
    open val height: Int
    fun close()
}
```

```kotlin
public external interface ImageBitmapOptions {
    var imageOrientation: ImageOrientation? /* = ImageOrientation.NONE */
        get() = definedExternally
        set(value) = definedExternally
    var premultiplyAlpha: PremultiplyAlpha? /* = PremultiplyAlpha.DEFAULT */
        get() = definedExternally
        set(value) = definedExternally
    var colorSpaceConversion: ColorSpaceConversion? /* = ColorSpaceConversion.DEFAULT */
        get() = definedExternally
        set(value) = definedExternally
    var resizeWidth: Int?
        get() = definedExternally
        set(value) = definedExternally
    var resizeHeight: Int?
        get() = definedExternally
        set(value) = definedExternally
    var resizeQuality: ResizeQuality? /* = ResizeQuality.LOW */
        get() = definedExternally
        set(value) = definedExternally
}
```

```kotlin
@kotlin.internal.InlineOnly
    val o = js("({})")
    o["imageOrientation"] = imageOrientation
    o["premultiplyAlpha"] = premultiplyAlpha
    o["colorSpaceConversion"] = colorSpaceConversion
    o["resizeWidth"] = resizeWidth
    o["resizeHeight"] = resizeHeight
    return o
}
```

Exposes the JavaScript [MessageEvent](https://developer.mozilla.org/en/docs/Web/API/MessageEvent) to Kotlin

```kotlin
public external open class MessageEvent(type: String, eventInitDict: MessageEventInit = definedExternally) : Event {
    open val data: Any?
    open val origin: String
    open val lastEventId: String
    open val source: UnionMessagePortOrWindowProxy?
    open val ports: Array<MessagePort>

    fun initMessageEvent(type: String, bubbles: Boolean, cancelable: Boolean, data: Any?, origin: String, lastEventId: String, source: UnionMessagePortOrWindowProxy?, ports: Array<MessagePort>)
```

```kotlin
companion object {
    val NONE: Short = undefined
    val CAPTURING_PHASE: Short = undefined
    val AT_TARGET: Short = undefined
    val BUBBLING_PHASE: Short = undefined
    val CAPTURING_PHASE: Short = undefined
    val AT_TARGET: Short = undefined
    val BUBBLING_PHASE: Short = undefined
}
```

```kotlin
public external interface MessageEventInit : EventInit {
    var data: Any? /* = null */
        get() = undefined
        set(value) = undefined
    var origin: String? /* = "" */
        get() = undefined
        set(value) = undefined
    var lastEventId: String? /* = "" */
        get() = undefined
        set(value) = undefined
    var source: UnionMessagePortOrWindowProxy? /* = null */
        get() = undefined
        set(value) = undefined
    var ports: Array<MessagePort>? /* = arrayOf() */
        get() = undefined
        set(value) = undefined
}
```

```kotlin
@kotlin.internal.InlineOnly
public inline fun MessageEventInit(data: Any?, origin: String, lastEventId: String, source: UnionMessagePortOrWindowProxy?, ports: Array<MessagePort>)
```
false): MessageEventInit {\n  val o = \n\n  o[["data"] \n  = data\n  o[["origin"] \n  = origin\n  o[["lastEventId"] \n  = lastEventId\n  o[["source"] \n  = source\n  o[["ports"] \n  = ports\n  o[["bubbles"] \n  = bubbles\n  o[["cancelable"] \n  = cancelable\n  o[["composed"] \n  = composed\n  return o\n}\n\n  * Exposes the JavaScript
[EventSource](https://developer.mozilla.org/en/docs/Web/API/EventSource) to Kotlin

  */\n  *\n\n  * Exposes the JavaScript
[EventSource](https://developer.mozilla.org/en/docs/Web/API/EventSource) to Kotlin

  */
public external open class EventSource(url: String, eventSourceInitDict: EventSourceInit = definedExternally) : EventTarget {
  open val url: String
  open val withCredentials: Boolean
  open val readyState: Short
  var onopen: ((Event) -> dynamic)?
  var onmessage: ((MessageEvent) -> dynamic)?
  var onerror: ((Event) -> dynamic)?
  fun close(): CompanionObject {
    CompanionObject {
      CONNECTING: Short
      OPEN: Short
      CLOSED: Short
    }
  }

  public external interface EventSourceInit {
    var withCredentials: Boolean? /* = false */
    get() = definedExternally
    set(value) = definedExternally
    var code: Short? /* = 0 */
    get() = definedExternally
    set(value) = definedExternally
    var reason: String? /* = "" */
    get() = definedExternally
    set(value) = definedExternally
  }

  @kotlin.internal.InlineOnly
  public inline fun EventSourceInit(withCredentials: Boolean? = false): EventSourceInit {
    val o = \n    o[["withCredentials"] = withCredentials
    return o\n}\n\n  * Exposes the JavaScript
[WebSocket](https://developer.mozilla.org/en/docs/Web/API/WebSocket) to Kotlin

  */
public external open class WebSocket(url: String, protocols: dynamic = definedExternally) : EventTarget {
  open val url: String
  open val readyState: Short
  open val bufferedAmount: Number
  var onopen: ((Event) -> dynamic)?
  var onerror: ((Event) -> dynamic)?
  var onclose: ((Event) -> dynamic)?
  open val extensions: String
  open val protocol: String
  var onmessage: ((MessageEvent) -> dynamic)?
  var binaryType: BinaryType
  fun close(code: Short = definedExternally, reason: String = definedExternally)
  fun send(data: String)
  fun send(data: Blob)
  fun send(data: ArrayBuffer)
  fun send(data: ArrayBufferView)

  CompanionObject {
    CONNECTING: Short
    OPEN: Short
    CLOSING: Short
    CLOSED: Short
  }

  public external interface CloseEventInit : EventInit {
    var wasClean: Boolean? /* = false */
    get() = definedExternally
    set(value) = definedExternally
    var code: Short? /* = 0 */
    get() = definedExternally
    set(value) = definedExternally
    var reason: String? /* = "" */
    get() = definedExternally
    set(value) = definedExternally
  }

  @kotlin.internal.InlineOnly
    val o = \n    o[["wasClean"] = wasClean
    o[["code"] = code
    o[["reason"] = reason
    o[["bubbles"] = bubbles
    o[["cancelable"] = cancelable
    o[["composed"] = composed
    return o\n}\n\n  * Exposes the JavaScript
[MessageChannel](https://developer.mozilla.org/en/docs/Web/API/MessageChannel) to Kotlin

  */
public external open class MessageChannel {
  open val port1: MessagePort
  open val port2: MessagePort
}

  public external abstract class MessagePort : EventTarget, UnionMessagePortOrWindowProxy, UnionMessagePortOrServiceWorker, UnionClientOrMessagePortOrServiceWorker {
    open var onmessage: ((MessageEvent) -> dynamic)?
    fun postMessage(message: Any?, transfer: Array<dynamic> = definedExternally)
    fun start()
    fun close()

  public external open class BroadcastChannel(name: String) : EventTarget {
    open var name: String
    var onmessage: ((MessageEvent) -> dynamic)?
    fun postMessage(message: Any?)

  public external abstract class BroadcastChannel(name: String) : EventTarget {
    open var name: String
    var onmessage: ((MessageEvent) -> dynamic)?
    fun postMessage(message: Any?)

  public external abstract class BroadcastChannel(name: String) : EventTarget {
    open var name: String
    var onmessage: ((MessageEvent) -> dynamic)?
    fun postMessage(message: Any?)
  }

  public external open class WorkerGlobalScope : EventTarget, UnionMessagePortOrServiceWorker, UnionClientOrMessagePortOrServiceWorker {
    open var onmessage: ((MessageEvent) -> dynamic)?
    fun postMessage(message: Any?, transfer: Array<dynamic> = definedExternally)
    fun start()
    fun close()

  public external open class WorkerGlobalScope : EventTarget, UnionMessagePortOrServiceWorker, UnionClientOrMessagePortOrServiceWorker {
    open var onmessage: ((MessageEvent) -> dynamic)?
    fun postMessage(message: Any?, transfer: Array<dynamic> = definedExternally)
    fun start()
    fun close()

  public external open class WorkerGlobalScope : EventTarget, UnionMessagePortOrServiceWorker, UnionClientOrMessagePortOrServiceWorker {
    open var onmessage: ((MessageEvent) -> dynamic)?
    fun postMessage(message: Any?, transfer: Array<dynamic> = definedExternally)
    fun start()
    fun close()
external open class StorageEvent {
    type: String,
    eventInitDict: StorageEventInit = definedExternally)
    : Event {
    key: String?
    oldValue: String?
    newValue: String?
    url: String

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public external interface StorageEventInit : EventInit {
    key: String? /* = null */
        get() = definedExternally
        set(value) = definedExternally
    oldValue: String? /* = null */
        get() = definedExternally
        set(value) = definedExternally
    newValue: String? /* = null */
        get() = definedExternally
        set(value) = definedExternally
    url: String? /* = null */
        get() = definedExternally
        set(value) = definedExternally
    storageArea: Storage?

    companion object {
        val NONE
        val CAPTURING_PHASE
        val AT_TARGET
        val BUBBLING_PHASE
    }
}

@kotlin.internal.InlineOnly
public inline fun StorageEventInit(key: String? = null,
    oldValue: String? = null,
    newValue: String? = null,
    url: String? = "",
    storageArea: Storage? = null,
    bubbles: Boolean? = false,
    cancelable: Boolean? = false,
    composed: Boolean? = false): StorageEventInit {
    val o = js("({})")
    o["key"] = key
    o["oldValue"] = oldValue
    o["newValue"] = newValue
    o["url"] = url
    o["storageArea"] = storageArea
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}

public external abstract class HTMLAppletElement : HTMLElement {
    align: String
    alt: String
    archive: String
    code: String
    codeBase: String
    height: String
    hspace: Int
    name: String
    _object: String
    vspace: Int
    width: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}

/**
 * Exposes the JavaScript [HTMLMarqueeElement](https://developer.mozilla.org/en/docs/Web/API/HTMLMarqueeElement) to Kotlin
 */
public external abstract class HTMLMarqueeElement : HTMLElement {
    behavior: String
    bgColor: String
    direction: String
    height: String
    hspace: Int
    loop: Int
    scrollAmount: Int
    scrollDelay: Int
    trueSpeed: Boolean
    vspace: Int
    width: String
    onbounce: ((Event) -> dynamic)?
    onfinish: ((Event) -> dynamic)?
    onstart: ((Event) -> dynamic)?
    fun start()
    fun stop()

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
    }
}

/**
 * Exposes the JavaScript [HTMLFrameSetElement](https://developer.mozilla.org/en/docs/Web/API/HTMLFrameSetElement) to Kotlin
 */
public external abstract class HTMLFrameSetElement : HTMLElement, WindowEventHandlers {
    cols: String
    rows: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
    }
}
/**
 * Exposes the JavaScript HTMLFontElement to Kotlin
 */
public external abstract class HTMLFontElement : HTMLElement {
    open var color: String
    open var face: String
    open var size: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

public external interface External {
    fun AddSearchProvider()
    fun IsSearchProviderInstalled()
}

public external interface EventInit {
    var bubbles: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var cancelable: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var composed: Boolean? /* = false */
        get() = definedExternally

    @kotlin.internal.InlineOnly
    public inline fun EventInit(bubbles: Boolean? = false,
         cancelable: Boolean? = false, composed: Boolean? = false): EventInit {
        val o = js("({})")
        o["bubbles"] = bubbles
        o["cancelable"] = cancelable
        o["composed"] = composed
        return o
    }

    /**
     * Exposes the JavaScript [HTMLFontElement](https://developer.mozilla.org/en/docs/Web/API/HTMLFontElement) to Kotlin
     */
    public external abstract class HTMLFontElement : HTMLElement {
        open var color: String
        open var face: String
        open var size: String

        companion object {
            val ELEMENT_NODE: Short
            val ATTRIBUTE_NODE: Short
            val TEXT_NODE: Short
            val CDATA_SECTION_NODE: Short
            val ENTITY_REFERENCE_NODE: Short
            val ENTITY_NODE: Short
            val PROCESSING_INSTRUCTION_NODE: Short
            val COMMENT_NODE: Short
            val DOCUMENT_NODE: Short
            val DOCUMENT_TYPE_NODE: Short
            val DOCUMENT_FRAGMENT_NODE: Short
            val NOTATION_NODE: Short
            val DOCUMENT_POSITION_DISCONNECTED: Short
            val DOCUMENT_POSITION_PRECEDING: Short
            val DOCUMENT_POSITION_FOLLOWING: Short
            val DOCUMENT_POSITION_CONTAINS: Short
            val DOCUMENT_POSITION_CONTAINED_BY: Short
            val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
        }
    }
}

public external interface External {
    fun AddSearchProvider()
    fun IsSearchProviderInstalled()
}

public external interface EventInit {
    var bubbles: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var cancelable: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var composed: Boolean? /* = false */
        get() = definedExternally

    @kotlin.internal.InlineOnly
    public inline fun EventInit(bubbles: Boolean? = false,
         cancelable: Boolean? = false, composed: Boolean? = false): EventInit {
        val o = js("({})")
        o["bubbles"] = bubbles
        o["cancelable"] = cancelable
        o["composed"] = composed
        return o
    }

    /**
     * Exposes the JavaScript [HTMLFontElement](https://developer.mozilla.org/en/docs/Web/API/HTMLFontElement) to Kotlin
     */
    public external abstract class HTMLFontElement : HTMLElement {
        open var color: String
        open var face: String
        open var size: String

        companion object {
            val ELEMENT_NODE: Short
            val ATTRIBUTE_NODE: Short
            val TEXT_NODE: Short
            val CDATA_SECTION_NODE: Short
            val ENTITY_REFERENCE_NODE: Short
            val ENTITY_NODE: Short
            val PROCESSING_INSTRUCTION_NODE: Short
            val COMMENT_NODE: Short
            val DOCUMENT_NODE: Short
            val DOCUMENT_TYPE_NODE: Short
            val DOCUMENT_FRAGMENT_NODE: Short
            val NOTATION_NODE: Short
            val DOCUMENT_POSITION_DISCONNECTED: Short
            val DOCUMENT_POSITION_PRECEDING: Short
            val DOCUMENT_POSITION_FOLLOWING: Short
            val DOCUMENT_POSITION_CONTAINS: Short
            val DOCUMENT_POSITION_CONTAINED_BY: Short
            val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
        }
    }
}
JavaScript [CustomEvent](https://developer.mozilla.org/en/docs/Web/API/CustomEvent) to Kotlin

```kotlin
public open class CustomEvent(type: String, eventInitDict: CustomEventInit = definedExternally) : Event {
  open var detail: Any?
  fun initCustomEvent(type: String, bubbles: Boolean, cancelable: Boolean, detail: Any?)
      companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public external interface CustomEventInit : EventInit {
    var detail: Any? /* = null */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["detail"] = detail
    o["bubbles"] = bubbles
    o["Cancelable"] = cancelable
    return o
}

public external interface EventListenerOptions {
    var capture: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun EventListenerOptions(capture: Boolean? = false): EventListenerOptions {
    val o = js("({})")
    o["capture"] = capture
    return o
}

public external interface AddEventListenerOptions : EventListenerOptions {
    var once: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun AddEventListenerOptions(once: Boolean? = false, capture: Boolean? = false): AddEventListenerOptions {
    val o = js("({})")
    o["once"] = once
    o["capture"] = capture
    return o
}

public external interface NonElementParentNode {
    fun getElementById(elementId: String): Element?
}

/**
 * Exposes the JavaScript [DocumentOrShadowRoot](https://developer.mozilla.org/en/docs/Web/API/DocumentOrShadowRoot) to Kotlin
 */
public external interface DocumentOrShadowRoot {
    val fullscreenElement: Element?
        get() = definedExternally
}

/**
 * Exposes the JavaScript [ParentNode](https://developer.mozilla.org/en/docs/Web/API/ParentNode) to Kotlin
 */
public external interface ParentNode {
    val children: HTMLCollection
    val firstElementChild: Element?
        get() = definedExternally
    val lastElementChild: Element?
        get() = definedExternally
    val childElementCount: Int
    fun prepend(vararg nodes: dynamic)
    fun append(vararg nodes: dynamic)
    fun querySelector(selectors: String): Element?
    fun querySelectorAll(selectors: String): NodeList
}

/**
 * Exposes the JavaScript [NonDocumentTypeChildNode](https://developer.mozilla.org/en/docs/Web/API/NonDocumentTypeChildNode) to Kotlin
 */
public external interface NonDocumentTypeChildNode {
    val previousElementSibling: Element?
        get() = definedExternally
    val nextElementSibling: Element?
        get() = definedExternally
}

/**
 * Exposes the JavaScript [ChildNode](https://developer.mozilla.org/en/docs/Web/API/ChildNode) to Kotlin
 */
public external interface ChildNode {
    fun before(vararg nodes: dynamic)
    fun after(vararg nodes: dynamic)
    fun replaceWith(vararg nodes: dynamic)
    fun remove()
}

/**
 * Exposes the JavaScript [Slotable](https://developer.mozilla.org/en/docs/Web/API/Slotable) to Kotlin
 */
public external interface Slotable {
    val assignedSlot: HTMLSlotElement?
        get() = definedExternally
}

/**
 * Exposes the JavaScript [NodeList](https://developer.mozilla.org/en/docs/Web/API/NodeList) to Kotlin
 */
public external abstract class NodeList : ItemArrayLike<Node> {
    override fun item(index: Int): Node?
}

@kotlin.internal.InlineOnly

/**
 * Exposes the JavaScript [HTMLCollection](https://developer.mozilla.org/en/docs/Web/API/HTMLCollection) to Kotlin
 */
public external abstract class HTMLCollection : ItemArrayLike<Element>, UnionElementOrHTMLCollection {
    override fun item(index: Int): Element?
    fun namedItem(name: String): Element?
}

@kotlin.internal.InlineOnly
public inline operator fun HTMLCollection.get(index: Int): Element? = asDynamic()[index]

/**
 * Exposes the JavaScript [MutationObserver](https://developer.mozilla.org/en/docs/Web/API/MutationObserver) to Kotlin
 */
public external open class MutationObserver(callback: (Array<MutationRecord>, MutationObserver) -> Unit) {
    fun
```
observe(target: Node, options: MutationObserverInit = definedExternally)
    fun disconnect()
    fun
    takeRecords(): Array<MutationRecord>
}

/**
 * Exposes the JavaScript [MutationObserverInit](https://developer.mozilla.org/en/docs/Web/API/MutationObserverInit) to Kotlin
 */
public external interface MutationObserverInit {
    var childList: Boolean? /* = false */
        get() =
        definedExternally
        set(value) = definedExternally
    var attributes: Boolean? /* = false */
        get() =
        definedExternally
        set(value) = definedExternally
    var characterData: Boolean?
        get() =
        definedExternally
        set(value) = definedExternally
    var subtree: Boolean? /* = false */
        get() =
        definedExternally
        set(value) = definedExternally
    var attributeOldValue: Boolean?
        get() =
        definedExternally
        set(value) = definedExternally
    var characterDataOldValue: Boolean?
        get() =
        definedExternally
        set(value) = definedExternally
    var attributeFilter: Array<String>?
        get() =
        definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
        o["childList"] = childList
        o["attributes"] = attributes
        o["characterData"] = characterData
        o["subtree"] = subtree
        o["attributeOldValue"] = attributeOldValue
        o["characterDataOldValue"] = characterDataOldValue
        o["attributeFilter"] = attributeFilter
        return o
}

/**
 * Exposes the JavaScript [MutationRecord](https://developer.mozilla.org/en/docs/Web/API/MutationRecord) to Kotlin
 */
public external abstract class MutationRecord {
    open val type: String
    open val target: Node
    open val addedNodes: NodeList
    open val removedNodes: NodeList
    open val previousSibling: Node?
    open val nextSibling: Node?
    open val attributeName: String?
    open val attributeNamespace: String?
    open val oldValue: String?
}

/**
 * Exposes the JavaScript [Node](https://developer.mozilla.org/en/docs/Web/API/Node) to Kotlin
 */
public external abstract class Node : EventTarget {
    open val nodeType: Short
    open val nodeName: String
    open val baseURI: String
    open val isConnected: Boolean
    open val ownerDocument: Document?
    open val parentNode: Node?
    open val parentElement: Element?
    open val childNodes: NodeList
    open val firstChild: Node?
    open val lastChild: Node?
    open val previousSibling: Node?
    open val nextSibling: Node?
    open var nodeValue: String?
    open var textContent: String?
    fun
    getRootNode(options: GetRootNodeOptions = definedExternally): Node?
    fun hasChildNodes(): Boolean
    fun
    normalize()
    fun
    cloneNode(deep: Boolean = definedExternally): Node
    fun replaceChild(node: Node, child: Node): Node
    fun addChild(node: Node): Node
    fun removeChild(child: Node): Node

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}

public external interface GetRootNodeOptions {
    var composed: Boolean? /* = false */
        get() =
        definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun GetRootNodeOptions(composed: Boolean? = false): GetRootNodeOptions {
    val o = js("({})")
        o["composed"] = composed
    return o
}

/**
 * Exposes the JavaScript [Document](https://developer.mozilla.org/en/docs/Web/API/Document) to Kotlin
 */
public external open class Document : Node, GlobalEventHandlers, DocumentAndElementEventHandlers,
override var onpointerup: ((PointerEvent) -> dynamic)?
override var onpointercancel: ((PointerEvent) -> dynamic)?
override var onpointerover: ((PointerEvent) -> dynamic)?
override var onpointerout: ((PointerEvent) -> dynamic)?
override var onpointerenter: ((PointerEvent) -> dynamic)?
override var onpointerleave: ((PointerEvent) -> dynamic)?
override var oncut: ((ClipboardEvent) -> dynamic)?
override var onpaste: ((ClipboardEvent) -> dynamic)?
override val fullscreenElement: Element?

fun getElementsByTagName(qualifiedName: String): HTMLCollection
fun getElementsByTagNameNS(namespace: String?, localName: String): HTMLCollection
fun getElementsByClassName(classNames: String): HTMLCollection
fun createElement(localName: String, options: ElementCreationOptions = definedExternally): Element
fun createElementNS(namespace: String?, qualifiedName: String, options: ElementCreationOptions = definedExternally): Element
fun createDocumentFragment(): DocumentFragment
fun createTextNode(data: String): Text
fun createCDATASection(data: String): CDATASection
fun createProcessingInstruction(target: String, data: String): ProcessingInstruction
fun importNode(node: Node, deep: Boolean = definedExternally): Node
fun adoptNode(node: Node): Node
fun createAttribute(localName: String): Attr
fun createAttributeNS(namespace: String?, qualifiedName: String): Attr
fun createEvent(`interface`: String): Event
fun createRange(): Range
fun getElementsByName(elementName: String): NodeList
fun open(type: String = definedExternally, replace: String = definedExternally): Document
fun open(url: String, name: String, features: String): Window
fun close()
fun write(vararg text: String)
fun writeln(vararg text: String)
fun hasFocus(): Boolean
fun execCommand(commandId: String, showUI: Boolean = definedExternally, value: String = definedExternally): Boolean

companion object {
val ELEMENT_NODE: Short
val ATTRIBUTE_NODE: Short
val TEXT_NODE: Short
val CDATA_SECTION_NODE: Short
val ENTITY_REFERENCE_NODE: Short
val ENTITY_NODE: Short
val DOCUMENT_NODE: Short
val DOCUMENT_TYPE_NODE: Short
val NOTATION_NODE: Short

fun elementFromPoint(x: Double, y: Double): Element?
fun elementsFromPoint(x: Double, y: Double): Array<Element>
fun caretPositionFromPoint(x: Double, y: Double): CaretPosition?
fun createTouchList(vararg touches: Touch): TouchList
fun exitFullscreen(): Promise<Unit>
fun getElementById(elementId: String): Element?
fun prepend(vararg nodes: dynamic)
fun append(vararg nodes: dynamic)
fun querySelector(selectors: String): Element?
fun querySelectorAll(selectors: String): Array<Element>
fun getBoxQuads(options: BoxQuadOptions /* = definedExternally */): Array<DOMQuad>
fun convertRectFromNode(rect: DOMRectReadOnly, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMQuad
fun convertQuadFromNode(quad: dynamic, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMQuad
fun convertPointFromNode(point: DOMPointInit, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMPoint
fun getBoxQuads(options: BoxQuadOptions /* = definedExternally */): Array<DOMQuad>
fun convertRectFromNode(rect: DOMRectReadOnly, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMQuad
fun convertQuadFromNode(quad: dynamic, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMQuad
fun convertPointFromNode(point: DOMPointInit, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMPoint
fun companion object {
val ELEMENT_NODE: Short
val ATTRIBUTE_NODE: Short
val TEXT_NODE: Short
val CDATA_SECTION_NODE: Short
val ENTITY_REFERENCE_NODE: Short
val ENTITY_NODE: Short
val DOCUMENT_NODE: Short
val DOCUMENT_TYPE_NODE: Short
val NOTATION_NODE: Short
val
DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

}
}

@kotlin.internal.InlineOnly
public inline operator fun Document.get(name: String): dynamic =
asDynamic()\[name\]

/**
 * Exposes the JavaScript
 [XMLDocument](https://developer.mozilla.org/en/docs/Web/API/XMLDocument) to Kotlin
 */
public external
open class XMLDocument : Document {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

public external interface ElementCreationOptions {
    var `is`: String?
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun ElementCreationOptions(`is`: String? = undefined): ElementCreationOptions {
    val o = js("({})")
    o["is"] = `is`
    return o
}

/**
 * Exposes
 the JavaScript
 [DOMImplementation](https://developer.mozilla.org/en/docs/Web/API/DOMImplementation) to
 Kotlin
 */
public external abstract class DOMImplementation {
    fun createDocumentType(qualifiedName: String, publicId: String, systemId: String): DocumentType
    fun createDocument(namespace: String?, qualifiedName: String, doctype: DocumentType? = definedExternally): XMLDocument
    fun createHTMLDocument(title: String = definedExternally): Document
    fun hasFeature(): Boolean
}

/**
 * Exposes
 the JavaScript
 [DocumentType](https://developer.mozilla.org/en/docs/Web/API/DocumentType) to
 Kotlin
 */
public external abstract class DocumentType : Node, ChildNode {
    open val name: String
    open val publicId: String
    open val systemId: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes
 the JavaScript
 [DocumentFragment](https://developer.mozilla.org/en/docs/Web/API/DocumentFragment) to
 Kotlin
 */
public external open class DocumentFragment : Node, NonElementParentNode, ParentNode {
    override val children: HTMLCollection
    override val firstElementChild: Element?
    override val lastElementChild: Element?
    override val childElementCount: Int
    override fun getElementById(elementId: String): Element?
    override fun prepend(vararg nodes: dynamic)
    override fun append(vararg nodes: dynamic)
    override fun querySelector(selectors: String): Element?
    override fun querySelectorAll(selectors: String): NodeList

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

*/

Exposes the JavaScript [ShadowRoot](https://developer.mozilla.org/en/docs/Web/API/ShadowRoot) to Kotlin

```kotlin
public external open class ShadowRoot : DocumentFragment, DocumentOrShadowRoot {
    open val mode: ShadowRootMode
    open val host: Element
    override val fullscreenElement: Element?

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [Element](https://developer.mozilla.org/en/docs/Web/API/Element) to Kotlin
 */
public external abstract class Element : Node, ParentNode, NonDocumentTypeChildNode, ChildNode, Slotable, GeometryUtils,
UnionElementOrHTMLCollection, UnionElementOrRadioNodeList, UnionElementOrMouseEvent,
UnionElementOrProcessingInstruction {
    open val namespaceURI: String?
    open val prefix: String?
    open val localName: String
    open val tagName: String
    open var id: String
    open var className: String
    open val classList: DOMTokenList
    open var slot: String
    open val attributes: NamedNodeMap
    open val shadowRoot: ShadowRoot?
    open var scrollTop: Double
    open var scrollLeft: Double
    open val scrollWidth: Int
    open val scrollHeight: Int
    open val clientTop: Int
    open val clientLeft: Int
    open val clientWidth: Int
    open val clientHeight: Int
    open var innerHTML: String
    open var outerHTML: String

    fun hasAttributes(): Boolean
    fun getAttributeNames(): Array<String>
    fun getAttribute(qualifiedName: String): String?
    fun getAttributeNS(namespace: String?, localName: String): String?
    fun setAttribute(qualifiedName: String, value: String)
    fun setAttributeNS(namespace: String?, qualifiedName: String, value: String)
    fun removeAttribute(qualifiedName: String)
    fun removeAttributeNS(namespace: String?, localName: String)
    fun hasAttribute(qualifiedName: String): Boolean
    fun hasAttributeNS(namespace: String?, localName: String): Boolean
    fun getAttributeNode(qualifiedName: String): Attr?
    fun getAttributeNodeNS(namespace: String?, localName: String): Attr?
    fun setAttributeNode(attr: Attr): Attr?
    fun setAttributeNodeNS(attr: Attr): Attr?
    fun removeAttributeNode(attr: Attr): Attr?
    fun attachShadow(init: ShadowRootInit): ShadowRoot?
    fun closest(selectors: String): Element?
    fun matches(selectors: String): Boolean
    fun webkitMatchesSelector(selectors: String): Boolean
    fun getElementsByTagName(qualifiedName: String): HTMLCollection
    fun getElementsByClassName(classNames: String): HTMLCollection
    fun insertAdjacentElement(where: String, element: Element): Element?
    fun insertAdjacentText(where: String, data: String)
    fun getClientRects(): Array<DOMRect>
    fun getBoundingClientRect(): DOMRect
    fun scrollIntoView()
    fun scrollIntoView(arg: dynamic)
    fun scrollTo(options: ScrollToOptions = definedExternally)
    fun scrollTo(x: Double, y: Double)
    fun scrollBy(options: ScrollToOptions = definedExternally)
    fun scrollBy(x: Double, y: Double)
    fun insertAdjacentHTML(position: String, text: String)
    fun setPointerCapture(pointerId: Int)
    fun releasePointerCapture(pointerId: Int)
    fun hasPointerCapture(pointerId: Int): Boolean
    fun requestFullscreen(): Promise<Unit>
    fun companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
    }
```
BoxQuadOptions */ = definedExternally */): Array<DOMQuad>
override fun convertQuadFromNode(quad: dynamic, from: dynamic, options: ConvertCoordinateOptions */ = definedExternally */): DOMQuad
override fun convertRectFromNode(rect: DOMRectReadOnly, from: dynamic, options: ConvertCoordinateOptions */ = definedExternally */): DOMQuad
override fun convertPointFromNode(point: DOMPointInit, from: dynamic, options: ConvertCoordinateOptions */ = definedExternally */): DOMPoint
override fun before(vararg nodes: dynamic)
override fun after(vararg nodes: dynamic)
override fun replaceWith(vararg nodes: dynamic)
override fun remove()

companion object {
val ELEMENT_NODE: Short
val ATTRIBUTE_NODE: Short
val TEXT_NODE: Short
val CDATA_SECTION_NODE: Short
val ENTITY_REFERENCE_NODE: Short
val ENTITY_NODE: Short
val PROCESSING_INSTRUCTION_NODE: Short
val COMMENT_NODE: Short
val DOCUMENT_NODE: Short
val DOCUMENT_TYPE_NODE: Short
val DOCUMENT_FRAGMENT_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
}

/**
 * Exposes the JavaScript
[CDATASection](https://developer.mozilla.org/en/docs/Web/API/CDATASection) to Kotlin
 */
public external open class CDATASection : Text {
companion object {
val ELEMENT_NODE: Short
val ATTRIBUTE_NODE: Short
val TEXT_NODE: Short
val CDATA_SECTION_NODE: Short
val ENTITY_REFERENCE_NODE: Short
val ENTITY_NODE: Short
val PROCESSING_INSTRUCTION_NODE: Short
val COMMENT_NODE: Short
val DOCUMENT_NODE: Short
val DOCUMENT_TYPE_NODE: Short
val DOCUMENT_FRAGMENT_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
}

/**
 * Exposes the JavaScript
[ProcessingInstruction](https://developer.mozilla.org/en/docs/Web/API/ProcessingInstruction) to Kotlin
 */
public external abstract class ProcessingInstruction : CharacterData, LinkStyle,
UnionElementOrProcessingInstruction {
open val target: String
companion object {
val ELEMENT_NODE: Short
val ATTRIBUTE_NODE: Short
val TEXT_NODE: Short
val CDATA_SECTION_NODE: Short
val ENTITY_REFERENCE_NODE: Short
val ENTITY_NODE: Short
val PROCESSING_INSTRUCTION_NODE: Short
val COMMENT_NODE: Short
val DOCUMENT_NODE: Short
val DOCUMENT_TYPE_NODE: Short
val DOCUMENT_FRAGMENT_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
}

/**
 * Exposes the JavaScript
[Comment](https://developer.mozilla.org/en/docs/Web/API/Comment) to Kotlin
 */
public external open class Comment(data: String = definedExternally) : CharacterData {
override val previousElementSibling: Element?
override val nextElementSibling: Element?
override fun before(vararg nodes: dynamic)
override fun after(vararg nodes: dynamic)
override fun replaceWith(vararg nodes: dynamic)
override fun remove()

companion object {
val ELEMENT_NODE: Short
val ATTRIBUTE_NODE: Short
val TEXT_NODE: Short
val CDATA_SECTION_NODE: Short
val ENTITY_REFERENCE_NODE: Short
val ENTITY_NODE: Short
val PROCESSING_INSTRUCTION_NODE: Short
val COMMENT_NODE: Short
val DOCUMENT_NODE: Short
val DOCUMENT_TYPE_NODE: Short
val DOCUMENT_FRAGMENT_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

/**
 * Exposes the JavaScript [Range](https://developer.mozilla.org/en/docs/Web/API/Range) to Kotlin
 */
public external open class Range {
  open val startContainer: Node
  open val startOffset: Int
  open val endContainer: Node
  open val endOffset: Int
  open val collapsed: Boolean
  open val commonAncestorContainer: Node
  fun setStart(node: Node, offset: Int)
  fun setEnd(node: Node, offset: Int)
  fun setStartBefore(node: Node)
  fun setStartAfter(node: Node)
  fun setEndBefore(node: Node)
  fun setEndAfter(node: Node)
  fun collapse(toStart: Boolean = definedExternally)
  fun selectNode(node: Node)
  fun selectNodeContents(node: Node)
  fun compareBoundaryPoints(how: Short, sourceRange: Range): Short
  fun deleteContents()
  fun extractContents(): DocumentFragment
  fun cloneContents(): DocumentFragment
  fun insertNode(node: Node)
  fun surroundContents(newParent: Node): Node?
  fun cloneRange(): Range
  fun detach()
  fun isPointInRange(node: Node, offset: Int): Boolean
  fun comparePoint(node: Node, offset: Int): Short
  fun intersectsNode(node: Node): Boolean
  fun getClientRects(): Array<DOMRect>
  fun getBoundingClientRect(): DOMRect
  fun createContextualFragment(fragment: String): DocumentFragment

  companion object {
    val START_TO_START: Short
    val START_TO_END: Short
    val END_TO_END: Short
    val END_TO_START: Short
  }
}

/**
 * Exposes the JavaScript [NodeIterator](https://developer.mozilla.org/en/docs/Web/API/NodeIterator) to Kotlin
 */
public external abstract class NodeIterator {
  open val root: Node
  open val referenceNode: Node
  open val pointerBeforeReferenceNode: Boolean
  open val whatToShow: Int
  open val filter: NodeFilter?
  fun nextNode(): Node?
  fun previousNode(): Node?
  fun detach()
}

/**
 * Exposes the JavaScript [TreeWalker](https://developer.mozilla.org/en/docs/Web/API/TreeWalker) to Kotlin
 */
public external abstract class TreeWalker {
  open val root: Node
  open val whatToShow: Int
  open val filter: NodeFilter?
  open var currentNode: Node
  fun parentNode(): Node?
  fun firstChild(): Node?
  fun lastChild(): Node?
  fun previousSibling(): Node?
  fun nextSibling(): Node?
  fun previousNode(): Node?
  fun nextNode(): Node?
}

/**
 * Exposes the JavaScript [NodeFilter](https://developer.mozilla.org/en/docs/Web/API/NodeFilter) to Kotlin
 */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface NodeFilter {
  fun acceptNode(node: Node): Short

  companion object {
    val FILTER_ACCEPT: Short
    val FILTER_REJECT: Short
    val FILTER_SKIP: Short
    val SHOW_ALL: Int
    val SHOW_ELEMENT: Int
    val SHOW_ATTRIBUTE: Int
    val SHOW_TEXT: Int
    val SHOW_CDATA_SECTION: Int
    val SHOW_ENTITY_REFERENCE: Int
    val SHOW_ENTITY: Int
    val SHOW_PROCESSING_INSTRUCTION: Int
    val SHOW_COMMENT: Int
    val SHOW_DOCUMENT: Int
    val SHOW_DOCUMENT_TYPE: Int
    val SHOW_DOCUMENT_FRAGMENT: Int
    val SHOW_NOTIFICATION: Int
    val SHOW_DATA: Int
    val SHOW_ENTITY_REFERENCE: Int
    val SHOW_ENTITY: Int
    val SHOW_ICON: Int
    val SHOW_SVG: Int
    val SHOW_HMENU: Int
    val SHOW_HMENU_ITEM: Int
    val SHOW_HMENU_SEPARATOR: Int
    val SHOW_HMENU_SEPARATOR_ITEM: Int
    val SHOW_HMENU_SEPARATOR_ITEM_SIMPLE: Int
    val SHOW_HMENU_SEPARATOR_ITEM_TREE: Int
    val SHOW_HMENU_SEPARATOR_ITEM_TREE_SIMPLE: Int
    val SHOW_HMENU_SEPARATOR_ITEM_TREE_SIMPLE_TREE: Int
    val SHOW_HMENU_SEPARATOR_ITEM_TREE_SIMPLE_TREE_SIMPLE: Int
    val SHOW_HMENU_SEPARATOR_ITEM_TREE_SIMPLE_TREE_SIMPLE_TREE: Int
    val SHOW_HMENU_SEPARATOR_ITEM_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE: Int
    val SHOW_HMENU_SEPARATOR_ITEM_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE: Int
    val SHOW_HMENU_SEPARATOR_ITEM_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE: Int
    val SHOW_HMENU_SEPARATOR_ITEM_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE: Int
    val SHOW_HMENU_SEPARATOR_ITEM_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE: Int
    val SHOW_HMENU_SEPARATOR_ITEM_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE: Int
    val SHOW_HMENU_SEPARATOR_ITEM_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE: Int
    val SHOW_HMENU_SEPARATOR_ITEM_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE: Int
    val SHOW_HMENU_SEPARATOR_ITEM_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_TREE: Int
    val SHOW_HMENU_SEPARATOR_ITEM_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_TREE_TREE: Int
    val SHOW_HMENU_SEPARATOR_ITEM_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_TREE_TREE_TREE: Int
    val SHOW_HMENU_SEPARATOR_ITEM_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_TREE_TREE_TREE_TREE: Int
    val SHOW_HMENU_SEPARATOR_ITEM_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_TREE_TREE_TREE_TREE_TREE: Int
    val SHOW_HMENU_SEPARATOR_ITEM_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_TREE_TREE_TREE_TREE_TREE_TREE: Int
    val SHOW_HMENU_SEPARATOR_ITEM_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_TREE_TREE_TREE_TREE_TREE_TREE_TREE: Int
    val SHOW_HMENU_SEPARATOR_ITEM_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_TREE_TREE_TREE_TREE_TREE_TREE_TREE_TREE: Int
    val SHOW_HMENU_SEPARATOR_ITEM_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_TREE_TREE_TREE_TREE_TREE_TREE_TREE_TREE_TREE: Int
    val SHOW_HMENU_SEPARATOR_ITEM_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_TREE_TREE_TREE_TREE_TREE_TREE_TREE_TREE_TREE_TREE: Int
    val SHOW_HMENU_SEPARATOR_ITEM_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_TREE_TREE_TREE_TREE_TREE_TREE_TREE_TREE_TREE_TREE_TREE: Int
    val SHOW_HMENU_SEPARATOR_ITEM_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_TREE_TREE_TREE_TREE_TREE_TREE_TREE_TREE_TREE_TREE_TREE_TREE: Int
    val SHOW_HMENU_SEPARATOR_ITEM_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_TREE_TREE_TREE_TREE_TREE_TREE_TREE_TREE_TREE_TREE_TREE_TREE_TREE: Int
    val SHOW_HMENU_SEPARATOR_ITEM_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_SIMPLE_TREE_TREE_TREE_TREE_TREE_TREE_TREE_TREE_TREE_TREE_TREE_TREE_TREE_TREE_TREE: Int
  }
}

/**
 * Exposes the JavaScript [DOMTokenList](https://developer.mozilla.org/en/docs/Web/API/DOMTokenList) to Kotlin
 */
public external abstract class DOMTokenList : ItemArrayLike<String> {
  open var value: String
  fun contains(token: String): Boolean
  fun add(vararg tokens: String)
  fun remove(vararg tokens: String)
  fun toggle(token: String, force: Boolean = definedExternally): Boolean
  fun replace(token: String, newToken: String)
  fun supports(token: String): Boolean

  override fun item(index: Int): String?
}

@kotlin.internal.InlineOnly
public inline operator fun DOMTokenList.get(index: Int): String? = asDynamic()[index]

/**
 * Exposes the JavaScript [DOMPointReadOnly](https://developer.mozilla.org/en/docs/Web/API/DOMPointReadOnly) to Kotlin
 */
public external open class DOMPointReadOnly(x: Double, y: Double, z: Double, w: Double) {
  open val x: Double
  open val y: Double
  open val z: Double
  open val w: Double
  fun matrixTransform(matrix: DOMMatrixReadOnly): DOMPoint
}

/**
 * Exposes the JavaScript [DOMPoint](https://developer.mozilla.org/en/docs/Web/API/DOMPoint) to Kotlin
 */
public external open class
DOMPoint : DOMPointReadOnly { constructor(point: DOMPointInit)
constructor(x: Double = definedExternally, y: Double = definedExternally, z: Double = definedExternally, w: Double = definedExternally)
override var x: Double
override var y: Double
override var z: Double
override var w: Double
} * Exposes the JavaScript

[DOMPointInit](https://developer.mozilla.org/en/docs/Web/API/DOMPointInit) to Kotlin

// Exposes the JavaScript

DOMPointInit { x = 0.0, y = 0.0, z = 0.0, w = 1.0 } * Exposes the JavaScript

DOMRect : DOMRect { x: Double = definedExternally, y: Double = definedExternally, width: Double = definedExternally, height: Double = definedExternally }
override var x: Double
override var y: Double
override var width: Double
override var height: Double

DOMRect { x = 0.0, y = 0.0, width = 0.0, height = 0.0 } * Exposes the JavaScript

DOMRect(x: Double = definedExternally, y: Double = definedExternally, width: Double = definedExternally, height: Double = definedExternally) { x = 0.0, y = 0.0, width = 0.0, height = 0.0 } * Exposes the JavaScript

DOMRectList : ItemArrayLike<DOMRect> { item(index: Int): DOMRect? } * Exposes the JavaScript

DOMMatrixReadOnly : DOMMatrixReadOnly { a: Double, b: Double, c: Double, d: Double, e: Double, f: Double, m11: Double, m12: Double, m13: Double, m14: Double, m21: Double, m22: Double, m23: Double, m24: Double, m31: Double, m32: Double, m33: Double, m34: Double, m41: Double, m42: Double, m43: Double, m44: Double, is2D: Boolean, isIdentity: Boolean, translate(tx: Double, ty: Double, tz: Double), scale(scale: Double, originX: Double, originY: Double), scale3d(scale: Double, originX: Double, originY: Double, originZ: Double), rotate(tx: Double, ty: Double, tz: Double) } * Exposes the JavaScript
Double = definedExternally, originY: Double = definedExternally): DOMMatrix\n  fun rotateFromVector(x: Double, y: Double): DOMMatrix\n  fun rotateAxisAngle(x: Double, y: Double, angle: Double): DOMMatrix\n  fun skewX(x: Double): DOMMatrix\n  fun skewY(y: Double): DOMMatrix\n  fun multiply(other: DOMMatrix): DOMMatrix\n  fun flipX(): DOMMatrix\n  fun flipY(): DOMMatrix\n  fun inverse(): DOMMatrix\n  fun transformPoint(point: DOMPointInit = definedExternally): DOMPoint\n  fun toFloat32Array(): Float32Array\n  fun toFloat64Array(): Float64Array\n
/**
 * Exposes the JavaScript [DOMMatrix](https://developer.mozilla.org/en/docs/Web/API/DOMMatrix) to Kotlin
 */
public external open class DOMMatrix() : DOMMatrixReadOnly {
  constructor(transformList: String)\n  constructor(other: DOMMatrixReadOnly)\n  constructor(array32: Float32Array)\n  constructor(array64: Float64Array)\n  constructor(numberSequence: Array<Double>)\n  override var a: Double\n  override var b: Double\n  override var c: Double\n  override var d: Double\n  override var e: Double\n  override var f: Double\n  override var m11: Double\n  override var m12: Double\n  override var m13: Double\n  override var m14: Double\n  override var m21: Double\n  override var m22: Double\n  override var m23: Double\n  override var m24: Double\n  override var m31: Double\n  override var m32: Double\n  override var m33: Double\n  override var m34: Double\n  override var m41: Double\n  override var m42: Double\n  override var m43: Double\n  override var m44: Double\n  fun multiplySelf(other: DOMMatrix): DOMMatrix\n  fun preMultiplySelf(other: DOMMatrix): DOMMatrix\n  fun translateSelf(tx: Double, ty: Double, tz: Double = definedExternally): DOMMatrix\n  fun scaleSelf(scale: Double, originX: Double = definedExternally, originY: Double = definedExternally): DOMMatrix\n  fun scaleSelf(scale: Double, originX: Double = definedExternally, originY: Double = definedExternally, originZ: Double = definedExternally): DOMMatrix\n  fun rotateSelf(angle: Double, originX: Double, originY: Double = definedExternally, originZ: Double = definedExternally): DOMMatrix\n  fun rotateFromVectorSelf(x: Double, y: Double): DOMMatrix\n  fun rotateAxisAngleSelf(x: Double, y: Double, z: Double, angle: Double): DOMMatrix\n  fun skewXSelf(sx: Double): DOMMatrix\n  fun skewYSelf(sy: Double): DOMMatrix\n  fun invertSelf(): DOMMatrix\n  fun setMatrixValue(transformList: String)\n}

public external interface ScrollOptions {
  var behavior: ScrollBehavior? /* = ScrollBehavior.AUTO */
      get() = definedExternally
      set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun ScrollOptions(behavior: ScrollBehavior? = ScrollBehavior.AUTO): ScrollOptions {
  val o = js("({})")
  o["behavior"] = behavior
  return o
}

/**
 * Exposes the JavaScript [ScrollToOptions](https://developer.mozilla.org/en/docs/Web/API/ScrollToOptions) to Kotlin
 */
public external interface ScrollToOptions : ScrollOptions {
  var left: Double?
      get() = definedExternally
      set(value) = definedExternally
  var top: Double?
      get() = definedExternally
      set(value) = definedExternally
}

@kotlin.internal.InlineOnly
  val o = js("({})")
  o["left"] = left
  o["top"] = top
  o["behavior"] = behavior
  return o
}

/**
 * Exposes the JavaScript [ MediaQueryList](https://developer.mozilla.org/en/docs/Web/API/MediaQueryList) to Kotlin
 */
public external abstract class MediaQueryList : EventTarget {
  open val media: String
  open val matches: Boolean
  open var onchange: ((Event) -> dynamic)?
  fun addListener(listener: EventListener?)
  fun addListener(listener: ((Event) -> Unit)?)
  fun removeListener(listener: EventListener?)
  fun removeListener(listener: ((Event) -> Unit)?)\n}

/**
 * Exposes the JavaScript [ MediaQueryListEvent](https://developer.mozilla.org/en/docs/Web/API/MediaQueryListEvent) to Kotlin
 */
public external open class MediaQueryListEvent(type: String, eventInitDict: MediaQueryListEventInit = definedExternally) : Event {
  open val media: String
  open val matches: Boolean

  companion object {
    val NONE: Short
    val CAPTURING_PHASE: Short
    val AT_TARGET: Short
    val BUBBLING_PHASE: Short
  }
}

public external interface MediaQueryListEventInit : EventInit {
  var media: String? /* = "" */
      get() = definedExternally
      set(value) = definedExternally
  var matches: Boolean? /* = false */
      get() = definedExternally
      set(value) =
@kotlin.internal.InlineOnly
    val o = js("({})")
    o["media"] = media
    o["matches"] = matches
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}

/**
 * Exposes the JavaScript [Screen](https://developer.mozilla.org/en/docs/Web/API/Screen) to Kotlin
 */
public external abstract class Screen {
    open val availWidth: Int
    open val availHeight: Int
    open val width: Int
    open val height: Int
    open val colorDepth: Int
    open val pixelDepth: Int
}

/**
 * Exposes the JavaScript [CaretPosition](https://developer.mozilla.org/en/docs/Web/API/CaretPosition) to Kotlin
 */
public external abstract class CaretPosition {
    open val offsetNode: Node
    open val offset: Int
}

public external interface ScrollIntoViewOptions : ScrollOptions {
    var block: ScrollLogicalPosition? /* = ScrollLogicalPosition.CENTER */
        get() = definedExternally
        set(value) = definedExternally
    var inline: ScrollLogicalPosition? /* = ScrollLogicalPosition.CENTER */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["block"] = block
    o["inline"] = inline
    return o
}

public external interface BoxQuadOptions {
    var box: CSSBoxType? /* = CSSBoxType.BORDER */
        get() = definedExternally
        set(value) = definedExternally
    var relativeTo: dynamic
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun BoxQuadOptions(box: CSSBoxType? = CSSBoxType.BORDER, relativeTo: dynamic = undefined): BoxQuadOptions {
    val o = js("({})")
    o["box"] = box
    o["relativeTo"] = relativeTo
    return o
}

/**
 * Exposes the JavaScript [GeometryUtils](https://developer.mozilla.org/en/docs/Web/API/GeometryUtils) to Kotlin
 */
public external interface GeometryUtils {
    fun getBoxQuads(options: BoxQuadOptions = definedExternally): Array<DOMQuad>
    fun convertQuadFromNode(quad: dynamic, from: dynamic, options: ConvertCoordinateOptions = definedExternally): DOMQuad
    fun convertRectFromNode(rect: DOMRectReadOnly, from: dynamic, options: ConvertCoordinateOptions = definedExternally): DOMQuad
    fun convertPointFromNode(point: DOMPointInit, from: dynamic, options: ConvertCoordinateOptions = definedExternally): DOMPoint
}

/**
 * Exposes the JavaScript [Touch](https://developer.mozilla.org/en/docs/Web/API/Touch) to Kotlin
 */
public external open class Touch {
    open val identifier: Int
    open val target: EventTarget
    open val screenX: Int
    open val screenY: Int
    open val clientX: Int
    open val clientY: Int
    open val pageX: Int
    open val pageY: Int

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public external open class TouchEvent : UIEvent {
    open val touches: TouchList
    open val targetTouches: TouchList
    open val changedTouches: TouchList
    open val altKey: Boolean
    open val metaKey: Boolean
    open val ctrlKey: Boolean

    override val type: String?
}

/**
 * Exposes the JavaScript [Image](https://developer.mozilla.org/en/docs/Web/API/Image) to Kotlin
 */
public external open class Image(width: Int = definedExternally, height: Int = definedExternally) : HTMLImageElement {
    override var onabort: ((Event) -> dynamic)?
    override var onblur: ((FocusEvent) -> dynamic)?
    override var onerror: ((Event) -> dynamic)?
    override var onfocus: ((FocusEvent) -> dynamic)?
    override var onkeydown: ((KeyboardEvent) -> dynamic)?
    override var onkeypress: ((KeyboardEvent) -> dynamic)?
    override var onkeyup: ((KeyboardEvent) -> dynamic)?
    override var onmousedown: ((MouseEvent) -> dynamic)?
    override var onmousecancel: ((MouseEvent) -> dynamic)?
    override var onmouseleave: ((MouseEvent) -> dynamic)?
    override var onmouseover: ((MouseEvent) -> dynamic)?
    override var onmouseout: ((MouseEvent) -> dynamic)?
    override var onmouseup: ((MouseEvent) -> dynamic)?
    override var onmousewheel: ((MouseWheelEvent) -> dynamic)?
}
override var oncanplay: ((Event) -> dynamic)?
override var oncanplaythrough: ((Event) -> dynamic)?
override var onchange: ((Event) -> dynamic)?
override var oncontextmenu: ((MouseEvent) -> dynamic)?
override var oncuechange: ((Event) -> dynamic)?
override var ondurationchange: ((Event) -> dynamic)?
override var ondrag: ((DragEvent) -> dynamic)?
override var ondragend: ((DragEvent) -> dynamic)?
override var ondragenter: ((DragEvent) -> dynamic)?
override var ondragexit: ((DragEvent) -> dynamic)?
override var ondragleave: ((DragEvent) -> dynamic)?
override var ondragstart: ((DragEvent) -> dynamic)?
override var ondragover: ((DragEvent) -> dynamic)?
override var onend: ((Event) -> dynamic)?
override var onerror: ((dynamic, String, Int, Int, Any?) -> dynamic)?
override var onfocus: ((FocusEvent) -> dynamic)?
override var oninput: ((InputEvent) -> dynamic)?
override var oninvalid: ((Event) -> dynamic)?
override var onkeydown: ((KeyboardEvent) -> dynamic)?
override var onkeypress: ((KeyboardEvent) -> dynamic)?
override var onkeyup: ((KeyboardEvent) -> dynamic)?
override var ondblclick: ((MouseEvent) -> dynamic)?
override var onmousedown: ((MouseEvent) -> dynamic)?
override var onmouseenter: ((MouseEvent) -> dynamic)?
override var onmouseleave: ((MouseEvent) -> dynamic)?
override var onmousemove: ((MouseEvent) -> dynamic)?
override var onmouseout: ((MouseEvent) -> dynamic)?
override var onmouseover: ((MouseEvent) -> dynamic)?
override var onmouseup: ((MouseEvent) -> dynamic)?
override var onwheel: ((WheelEvent) -> dynamic)?
override var onpause: ((Event) -> dynamic)?
override var onplay: ((Event) -> dynamic)?
override var onplaying: ((Event) -> dynamic)?
override var onprogress: ((ProgressEvent) -> dynamic)?
override var onratechange: ((Event) -> dynamic)?
override var onreset: ((Event) -> dynamic)?
override var onresize: ((Event) -> dynamic)?
override var onscroll: ((Event) -> dynamic)?
override var onselect: ((Event) -> dynamic)?
override var onshow: ((Event) -> dynamic)?
override var onstalled: ((Event) -> dynamic)?
override var onsubmit: ((Event) -> dynamic)?
override var ontimeupdate: ((Event) -> dynamic)?
override var onvolumechange: ((Event) -> dynamic)?
override var onwaiting: ((Event) -> dynamic)?
override var onpointercapture: ((PointerEvent) -> dynamic)?
override var onpointerdown: ((PointerEvent) -> dynamic)?
override var onpointermove: ((PointerEvent) -> dynamic)?
override var onpointerup: ((PointerEvent) -> dynamic)?
override var onpointercancel: ((PointerEvent) -> dynamic)?
override var onpointerout: ((PointerEvent) -> dynamic)?
override var onpointerleave: ((PointerEvent) -> dynamic)?
override var oncopy: ((ClipboardEvent) -> dynamic)?
override var oncut: ((ClipboardEvent) -> dynamic)?
override var onpaste: ((ClipboardEvent) -> dynamic)?
override var contentEditable: Boolean
override var style: CSSStyleDeclaration
override var children: HTMLCollection
override var firstElementChild: Element?
override var lastElementChild: Element?
override var childElementCount: Int
override var previousElementSibling: Element?
override var nextElementSibling: Element?
override var assignedSlot: HTMLSlotElement?
override fun prepend(vararg nodes: dynamic): HTMLCollection
override fun append(vararg nodes: dynamic): HTMLCollection
override fun querySelector(selectors: String): Element?
override fun querySelectorAll(selectors: String): NodeList
override fun before(vararg nodes: dynamic): HTMLCollection
override fun after(vararg nodes: dynamic): HTMLCollection
override fun replaceWith(vararg nodes: dynamic): HTMLCollection
override fun remove(): HTMLCollection
override fun getBoxQuads(options: BoxQuadOptions /* = definedExternally */): Array<DOMQuad>
override fun convertQuadFromNode(quad: dynamic, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMQuad
override fun convertRectFromNode(rect: DOMRectReadOnly, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMQuad

dynamic)

override var onpaste: ((ClipboardEvent) -> dynamic)?

override var contentEditable: String?

override val isContentEditable: Boolean

override val style: CSSStyleDeclaration

override val children: HTMLCollection

override val firstElementChild: Element?

override val lastElementChild: Element?

override val childElementCount: Int

override val previousElementSibling: Element?

override val nextElementSibling: Element?

override val assignedSlot: HTMLSlotElement?

override fun prepend(vararg nodes: dynamic)

override fun append(vararg nodes: dynamic)

override fun querySelector(selectors: String): Element?

override fun querySelectorAll(selectors: String): NodeList

override fun before(vararg nodes: dynamic)

override fun after(vararg nodes: dynamic)

override fun replaceWith(vararg nodes: dynamic)

override fun remove()

override fun getBoxQuads(options: BoxQuadOptions /* = definedExternally */) :

Array<DOMQuad>

override fun convertQuadFromNode(quad: dynamic, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */) : DOMQuad

override fun convertRectFromNode(rect: DOMRectReadOnly, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */) : DOMPoint

override fun convertPointFromNode(point: DOMPointInit, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */) : DOMPoint

companion object {
    val NETWORK_EMPTY: Short

val NETWORK_IDLE: Short

val NETWORK_LOADING: Short

val NETWORK_NO_SOURCE: Short

val HAVE_NOTHING: Short

val HAVE_METADATA: Short

val HAVE_CURRENT_DATA: Short

val HAVE_FUTURE_DATA: Short

val HAVE_ENOUGH_DATA: Short

val ELEMENT_NODE: Short

val ATTRIBUTE_NODE: Short

val TEXT_NODE: Short

val CDATA_SECTION_NODE: Short

val ENTITY_REFERENCE_NODE: Short

val ENTITY_NODE: Short

val PROCESSING_INSTRUCTION_NODE: Short

val COMMENT_NODE: Short

val DOCUMENT_NODE: Short

val DOCUMENT_TYPE_NODE: Short

val DOCUMENT_FRAGMENT_NODE: Short

val NOTATION_NODE: Short

val DOCUMENT_POSITION_DISCONNECTED: Short

val DOCUMENT_POSITION_PRECEDING: Short

val DOCUMENT_POSITION_FOLLOWING: Short

val DOCUMENT_POSITION_CONTAINS: Short

val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
}

/**
 * Exposes the JavaScript Option object to Kotlin
 */
public external open class Option(text: String = definedExternally, value: String = definedExternally, defaultSelected: Boolean = definedExternally, selected: Boolean = definedExternally) : HTMLOptionElement {

override var onabort: ((Event) -> dynamic)?

override var onblur: ((FocusEvent) -> dynamic)?

override var oncancel: ((Event) -> dynamic)?

override var oncanplay: ((Event) -> dynamic)?

override var oncanplaythrough: ((Event) -> dynamic)?

override var onchange: ((Event) -> dynamic)?

override var onclick: ((MouseEvent) -> dynamic)?

override var onclose: ((Event) -> dynamic)?

override var oncontextmenu: ((MouseEvent) -> dynamic)?

override var oncuechange: ((Event) -> dynamic)?

override var ondblclick: ((MouseEvent) -> dynamic)?

override var ondrag: ((DragEvent) -> dynamic)?

override var ondragend: ((DragEvent) -> dynamic)?

override var ondragenter: ((DragEvent) -> dynamic)?

override var ondragexit: ((DragEvent) -> dynamic)?

override var ondragleave: ((DragEvent) -> dynamic)?

override var ondragstart: ((DragEvent) -> dynamic)?

override var ondragdata: ((DragEvent) -> dynamic)?

override var ondragover: ((DragEvent) -> dynamic)?

override var ondragstart: ((DragEvent) -> dynamic)?

override var onfocus: ((FocusEvent) -> dynamic)?

override var oninput: ((InputEvent) -> dynamic)?

override var oninvalid: ((Event) -> dynamic)?

override var onkeydown: ((KeyboardEvent) -> dynamic)?

override var onkeyup: ((KeyboardEvent) -> dynamic)?

override var onkeydown: ((KeyboardEvent) -> dynamic)?

override var onkeyup: ((KeyboardEvent) -> dynamic)?

override var onblur: ((FocusEvent) -> dynamic)?

override var oninput: ((InputEvent) -> dynamic)?

override var oninvalid: ((Event) -> dynamic)?

override var onkeydown: ((KeyboardEvent) -> dynamic)?

override var onkeyup: ((KeyboardEvent) -> dynamic)?

override var onblur: ((FocusEvent) -> dynamic)?

override var oninput: ((InputEvent) -> dynamic)?

override var oninvalid: ((Event) -> dynamic)?

override var onkeydown: ((KeyboardEvent) -> dynamic)?

override var onkeyup: ((KeyboardEvent) -> dynamic)?

override var onblur: ((FocusEvent) -> dynamic)?

override var oninput: ((InputEvent) -> dynamic)?

override var oninvalid: ((Event) -> dynamic)?

override var onkeydown: ((KeyboardEvent) -> dynamic)?

override var onkeyup: ((KeyboardEvent) -> dynamic)?


/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface CanvasLineCap {
  companion object
}

public inline val CanvasLineCap.Companion.BUTT: CanvasLineCap get() = \"butt\".asDynamic().unsafeCast<CanvasLineCap>()

public inline val CanvasLineCap.Companion.ROUND: CanvasLineCap get() = \"round\".asDynamic().unsafeCast<CanvasLineCap>()

public inline val CanvasLineCap.Companion.SQUARE: CanvasLineCap get() = \"square\".asDynamic().unsafeCast<CanvasLineCap>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface CanvasLineJoin {
  companion object
}

public inline val CanvasLineJoin.Companion.REVERSE: CanvasLineJoin get() = \"reverse\".asDynamic().unsafeCast<CanvasLineJoin>()

public inline val CanvasLineJoin.Companion.MITER: CanvasLineJoin get() = \"miter\".asDynamic().unsafeCast<CanvasLineJoin>()

public inline val CanvasLineJoin.Companion.BEVEL: CanvasLineJoin get() = \"bevel\".asDynamic().unsafeCast<CanvasLineJoin>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface CanvasTextAlign {
  companion object
}

public inline val CanvasTextAlign.Companion.START: CanvasTextAlign get() = \"start\".asDynamic().unsafeCast<CanvasTextAlign>()

public inline val CanvasTextAlign.Companion.END: CanvasTextAlign get() = \"end\".asDynamic().unsafeCast<CanvasTextAlign>()

public inline val CanvasTextAlign.Companion.RIGHT: CanvasTextAlign get() = \"right\".asDynamic().unsafeCast<CanvasTextAlign>()

public inline val CanvasTextAlign.Companion.LEFT: CanvasTextAlign get() = \"left\".asDynamic().unsafeCast<CanvasTextAlign>()

public inline val CanvasTextAlign.Companion.CENTER: CanvasTextAlign get() = \"center\".asDynamic().unsafeCast<CanvasTextAlign>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface CanvasTextBaseline {
  companion object
}

public inline val CanvasTextBaseline.Companion.TOP: CanvasTextBaseline get() = \"top\".asDynamic().unsafeCast<CanvasTextBaseline>()

public inline val CanvasTextBaseline.Companion.MIDDLE: CanvasTextBaseline get() = \"middle\".asDynamic().unsafeCast<CanvasTextBaseline>()

public inline val CanvasTextBaseline.Companion.ALBPHABETIC: CanvasTextBaseline get() = \"alphabetic\".asDynamic().unsafeCast<CanvasTextBaseline>()

public inline val CanvasTextBaseline.Companion.IDEOGRAPHIC: CanvasTextBaseline get() = \"ideographic\".asDynamic().unsafeCast<CanvasTextBaseline>()

public inline val CanvasTextBaseline.Companion.BOTTOM: CanvasTextBaseline get() = \"bottom\".asDynamic().unsafeCast<CanvasTextBaseline>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface CanvasDirection {
  companion object
}

public inline val CanvasDirection.Companion.LTR: CanvasDirection get() = \"ltr\".asDynamic().unsafeCast<CanvasDirection>()

public inline val CanvasDirection.Companion.RTL: CanvasDirection get() = \"rtl\".asDynamic().unsafeCast<CanvasDirection>()

public inline val CanvasDirection.Companion.INHERIT: CanvasDirection get() = \"inherit\".asDynamic().unsafeCast<CanvasDirection>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface ScrollRestoration {
  companion object
}

public inline val ScrollRestoration.Companion.AUTO: ScrollRestoration get() = \"auto\".asDynamic().unsafeCast<ScrollRestoration>()


/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface PremultiplyAlpha {
    companion object
}

public inline val PremultiplyAlpha.Companion.NONE: PremultiplyAlpha get() = "none".asDynamic().unsafeCast<PremultiplyAlpha>()

public inline val PremultiplyAlpha.Companion.PREMULTIPLY: PremultiplyAlpha get() = "premultiply".asDynamic().unsafeCast<PremultiplyAlpha>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface ColorSpaceConversion {
    companion object
}

public inline val ColorSpaceConversion.Companion.NONE: ColorSpaceConversion get() = "none".asDynamic().unsafeCast<ColorSpaceConversion>()

public inline val ColorSpaceConversion.Companion.DEFAULT: ColorSpaceConversion get() = "default".asDynamic().unsafeCast<ColorSpaceConversion>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface ResizeQuality {
    companion object
}

public inline val ResizeQuality.Companion.PIXELATED: ResizeQuality get() = "pixelated".asDynamic().unsafeCast<ResizeQuality>()

public inline val ResizeQuality.Companion.LOW: ResizeQuality get() = "low".asDynamic().unsafeCast<ResizeQuality>()

public inline val ResizeQuality.Companion.MEDIUM: ResizeQuality get() = "medium".asDynamic().unsafeCast<ResizeQuality>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface BinaryType {
    companion object
}

public inline val BinaryType.Companion.BLOB: BinaryType get() = "blob".asDynamic().unsafeCast<BinaryType>()

public inline val BinaryType.Companion.ARRAYBUFFER: BinaryType get() = "arraybuffer".asDynamic().unsafeCast<BinaryType>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface WorkerType {
    companion object
}

public inline val WorkerType.Companion.CLASSIC: WorkerType get() = "classic".asDynamic().unsafeCast<WorkerType>()

public inline val WorkerType.Companion.MODULE: WorkerType get() = "module".asDynamic().unsafeCast<WorkerType>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface ShadowRootMode {
    companion object
}

public inline val ShadowRootMode.Companion.OPEN: ShadowRootMode get() = "open".asDynamic().unsafeCast<ShadowRootMode>()

public inline val ShadowRootMode.Companion.CLOSED: ShadowRootMode get() = "closed".asDynamic().unsafeCast<ShadowRootMode>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface ScrollBehavior {
    companion object
}

public inline val ScrollBehavior.Companion.AUTO: ScrollBehavior get() = "auto".asDynamic().unsafeCast<ScrollBehavior>()

public inline val ScrollBehavior.Companion.INSTANT: ScrollBehavior get() = "instant".asDynamic().unsafeCast<ScrollBehavior>()

public inline val ScrollBehavior.Companion.SMOOTH: ScrollBehavior get() = "smooth".asDynamic().unsafeCast<ScrollBehavior>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface ScrollLogicalPosition {
    companion object
}


public inline val ScrollLogicalPosition.Companion.CENTER: ScrollLogicalPosition get() = "center".asDynamic().unsafeCast<ScrollLogicalPosition>()

headers = headers
body = body
referrer = referrer
referrerPolicy = referrerPolicy
mode = mode
credentials = credentials
cache = cache
referrerPolicy
mode
credentials
cache
referrerPolicy
mode
credentials
cache
referrerPolicy
mode
credentials

*Exposes the JavaScript [Response](https://developer.mozilla.org/en/docs/Web/API/Response) to Kotlin

**public external open class Response(body: dynamic = definedExternally, init: ResponseInit = definedExternally) : Body**

**fun clone(): Response**

**fun arrayBuffer(): Promise<ArrayBuffer>**

**fun blob(): Promise<Blob>**

**fun formData(): Promise<FormData>**

**fun json(): Promise<Any?>**

**fun text(): Promise<String>**

*please, don't implement this interface!*

```kotlin
@kotlin.internal.InlineOnly
    val o = js("({})")
    o["status"] = status
    o["statusText"] = statusText
    o["headers"] = headers
    return o
}
```
RequestRedirect.Companion.MANUAL: RequestRedirect get() =
"manual".asDynamic().unsafeCast<RequestRedirect>()
/* please, don't implement this interface! */
public external interface ResponseType {
    companion object
}

public inline val ResponseType.Companion.BASIC: ResponseType get() =
"basic".asDynamic().unsafeCast<ResponseType>()

public inline val ResponseType.Companion.CORS: ResponseType get() =
"cors".asDynamic().unsafeCast<ResponseType>()

public inline val ResponseType.Companion.DEFAULT: ResponseType get() =
"default".asDynamic().unsafeCast<ResponseType>()

public inline val ResponseType.Companion.ERROR: ResponseType get() =
"error".asDynamic().unsafeCast<ResponseType>()

public inline val ResponseType.Companion.OPAQUE: ResponseType get() =
"opaque".asDynamic().unsafeCast<ResponseType>()

/* Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
   Use of this source code is governed by the Apache 2.0 license that can be
   found in the license/LICENSE.txt file. */

// NOTE: THIS FILE IS AUTO-GENERATED, DO NOT EDIT!
// See github.com/kotlin/dukat for details
package org.w3c.dom.mediacapture

import kotlin.js.*
import org.khronos.webgl.*
import org.w3c.css.masking.*
import org.w3c.dom.*
import org.w3c.dom.clipboard.*
import org.w3c.dom.css.*
import org.w3c.dom.encryptedmedia.*
import org.w3c.dom.events.*
import org.w3c.dom.mediasource.*
import org.w3c.dom.parsing.*
import org.w3c.dom.pointerevents.*
import org.w3c.dom.svg.*
import org.w3c.dom.url.*
import org.w3c.fetch.*
import org.w3c.files.*
import org.w3c.notifications.*
import org.w3c.performance.*
import org.w3c.workers.*
import org.w3c.xhr.*

/**
 * Exposes the JavaScript [MediaStream](https://developer.mozilla.org/en/docs/Web/API/MediaStream) to Kotlin
 */
public external open class MediaStream() : EventTarget, MediaProvider {
    constructor(stream: MediaStream)
    constructor(tracks: Array<MediaStreamTrack>)
    open val id: String
    open val active: Boolean
    var onaddtrack: ((MediaStreamTrackEvent) -> dynamic)?
    var onremovetrack: ((MediaStreamTrackEvent) -> dynamic)?
    fun getAudioTracks(): Array<MediaStreamTrack>
    fun getVideoTracks(): Array<MediaStreamTrack>
    fun getTracks(): Array<MediaStreamTrack>
    fun getTrackById(trackId: String): MediaStreamTrack?
    fun addTrack(track: MediaStreamTrack)
    fun removeTrack(track: MediaStreamTrack)
    fun clone(): MediaStream
}

/**
 * Exposes the JavaScript [MediaStreamTrack](https://developer.mozilla.org/en/docs/Web/API/MediaStreamTrack) to Kotlin
 */
public external abstract class MediaStreamTrack : EventTarget {
    open val kind: String
    open val id: String
    open val label: String
    open var enabled: Boolean
    open var muted: Boolean
    open var onmute: ((Event) -> dynamic)?
    open var onunmute: ((Event) -> dynamic)?
    open val readyState: MediaStreamTrackState
    open var onended: ((Event) -> dynamic)?
    open var onoverconstrained: ((Event) -> dynamic)?
    fun clone(): MediaStreamTrack
    fun stop()
    fun getCapabilities(): MediaTrackCapabilities
    fun getConstraints(): MediaTrackConstraints
    fun applyConstraints(constraints: MediaTrackConstraints = definedExternally): Promise<Unit>
}

/**
 * Exposes the JavaScript [MediaTrackSupportedConstraints](https://developer.mozilla.org/en/docs/Web/API/MediaTrackSupportedConstraints) to Kotlin
 */
public external interface MediaTrackSupportedConstraints {
    var width: Boolean? /* = true */
        get() = definedExternally
        set(value) = definedExternally
    var height: Boolean? /* = true */
        get() = definedExternally
        set(value) = definedExternally
    var aspectRatio: Boolean? /* = true */
        get() = definedExternally
        set(value) = definedExternally
    var frameRate: Boolean? /* = true */
        get() = definedExternally
        set(value) = definedExternally
    var facingMode: Boolean? /* = true */
        get() = definedExternally
        set(value) = definedExternally
    var resizeMode: Boolean? /* = true */
        get() = definedExternally
        set(value) = definedExternally
    var volume: Boolean? /* = true */
        get() = definedExternally
        set(value) = definedExternally
    var sampleRate: Boolean? /* = true */
        get() = definedExternally
        set(value) = definedExternally
    var sampleSize: Boolean? /* = true */
        get() = definedExternally
        set(value) = definedExternally
}
```kotlin
definedExternally
set(value) = definedExternally
var echoCancellation: Boolean? /* = true */
get() = definedExternally
set(value) = definedExternally
var autoGainControl: Boolean? /* = true */
get() = definedExternally
set(value) = definedExternally
var noiseSuppression: Boolean? /* = true */
get() = definedExternally
set(value) = definedExternally
var latency: Boolean? /* = true */
get() = definedExternally
set(value) = definedExternally
var channelCount: Boolean? /* = true */
get() = definedExternally
set(value) = definedExternally
var deviceId: Boolean? /* = true */
get() = definedExternally
set(value) = definedExternally
var groupId: Boolean? /* = true */
```

```kotlin
get() = definedExternally
set(value) = definedExternally
var height: ULongRange? = undefined
get() = definedExternally
set(value) = definedExternally
var aspectRatio: DoubleRange? = undefined
get() = definedExternally
set(value) = definedExternally
var frameRate: DoubleRange? = undefined
get() = definedExternally
set(value) = definedExternally
var facingMode: Array<String>? = undefined
get() = definedExternally
set(value) = definedExternally
var resizeMode: Array<String>? = undefined
get() = definedExternally
set(value) = definedExternally
var volume: DoubleRange? = undefined
get() = definedExternally
set(value) = definedExternally
var sampleRate: ULongRange? = undefined
get() = definedExternally
set(value) = definedExternally
var sampleSize: ULongRange? = undefined
get() = definedExternally
set(value) = definedExternally
var echoCancellation: Array<Boolean>? = undefined
get() = definedExternally
set(value) = definedExternally
var autoGainControl: Array<Boolean>? = undefined
get() = definedExternally
set(value) = definedExternally
var noiseSuppression: Array<Boolean>? = undefined
get() = definedExternally
set(value) = definedExternally
var latency: DoubleRange? = undefined
get() = definedExternally
set(value) = definedExternally
var channelCount: ULongRange? = undefined
get() = definedExternally
set(value) = definedExternally
var deviceId: String? = undefined
get() = definedExternally
set(value) = definedExternally
var groupId: String? = undefined
get() = definedExternally
set(value) = definedExternally
```

```kotlin
/**
 * Exposes the JavaScript
[MediaTrackConstraints](https://developer.mozilla.org/en/docs/Web/API/MediaTrackConstraints) to Kotlin
```
*/
public external interface MediaTrackConstraints : MediaTrackConstraintSet {
    var advanced:
        get() = definedExternally
        set(value) = definedExternally
    var width: Int?
        get() = definedExternally
        set(value) = definedExternally
    var height: Int?
        get() = definedExternally
        set(value) = definedExternally
    var aspectRatio: Double?
        get() = definedExternally
        set(value) = definedExternally
    var frameRate: Int?
        get() = definedExternally
        set(value) = definedExternally
    var facingMode: String?
        get() = definedExternally
        set(value) = definedExternally
    var resizeMode: String?
        get() = definedExternally
        set(value) = definedExternally
    var deviceId: String?
        get() = definedExternally
        set(value) = definedExternally
    var groupId: String?
        get() = definedExternally
        set(value) = definedExternally
}

public external interface MediaTrackConstraintSet {
    var width: Int?
        get() = definedExternally
        set(value) = definedExternally
    var height: Int?
        get() = definedExternally
        set(value) = definedExternally
    var aspectRatio: Double?
        get() = definedExternally
        set(value) = definedExternally
    var frameRate: Int?
        get() = definedExternally
        set(value) = definedExternally
    var facingMode: String?
        get() = definedExternally
        set(value) = definedExternally
    var resizeMode: String?
        get() = definedExternally
        set(value) = definedExternally
    var deviceId: String?
        get() = definedExternally
        set(value) = definedExternally
    var groupId: String?
        get() = definedExternally
        set(value) = definedExternally
}

/**
 * Exposes the JavaScript
 * MediaTrackSettings to Kotlin
 */
public external interface MediaTrackSettings {
    var width: Int?
        get() = definedExternally
        set(value) = definedExternally
    var height: Int?
        get() = definedExternally
        set(value) = definedExternally
    var aspectRatio: Double?
        get() = definedExternally
        set(value) = definedExternally
    var frameRate: Int?
        get() = definedExternally
        set(value) = definedExternally
    var facingMode: String?
        get() = definedExternally
        set(value) = definedExternally
    var resizeMode: String?
        get() = definedExternally
        set(value) = definedExternally
    var deviceId: String?
        get() = definedExternally
        set(value) = definedExternally
    var groupId: String?
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun MediaTrackSettings.get() = definedExternally

@kotlin.internal.InlineOnly
public inline fun MediaTrackConstraints.set(value) = definedExternally

@kotlin.internal.InlineOnly
public inline fun MediaTrackConstraints.advanced.set(value) = definedExternally

@kotlin.internal.InlineOnly
public inline fun MediaTrackConstraints.width.set(value) = definedExternally

@kotlin.internal.InlineOnly
public inline fun MediaTrackConstraints.height.set(value) = definedExternally

@kotlin.internal.InlineOnly
public inline fun MediaTrackConstraints.aspectRatio.set(value) = definedExternally

@kotlin.internal.InlineOnly
public inline fun MediaTrackConstraints.frameRate.set(value) = definedExternally

@kotlin.internal.InlineOnly
public inline fun MediaTrackConstraints.facingMode.set(value) = definedExternally

@kotlin.internal.InlineOnly
public inline fun MediaTrackConstraints.resizeMode.set(value) = definedExternally

@kotlin.internal.InlineOnly
public inline fun MediaTrackConstraints.deviceId.set(value) = definedExternally

@kotlin.internal.InlineOnly
public inline fun MediaTrackConstraints.groupId.set(value) = definedExternally

@kotlin.internal.InlineOnly
public inline fun MediaTrackConstraints.volume.set(value) = definedExternally

@kotlin.internal.InlineOnly
public inline fun MediaTrackConstraints.sampleRate.set(value) = definedExternally

@kotlin.internal.InlineOnly
public inline fun MediaTrackConstraints.sampleSize.set(value) = definedExternally

@kotlin.internal.InlineOnly
public inline fun MediaTrackConstraints.echoCancellation.set(value) = definedExternally

@kotlin.internal.InlineOnly
public inline fun MediaTrackConstraints.autoGainControl.set(value) = definedExternally

@kotlin.internal.InlineOnly
public inline fun MediaTrackConstraints.noiseSuppression.set(value) = definedExternally

@kotlin.internal.InlineOnly
public inline fun MediaTrackConstraints.latency.set(value) = definedExternally

@kotlin.internal.InlineOnly
public inline fun MediaTrackConstraints.channelCount.set(value) = definedExternally

@kotlin.internal.InlineOnly
public inline fun MediaTrackSettings.width.set(value) = definedExternally

@kotlin.internal.InlineOnly
public inline fun MediaTrackSettings.height.set(value) = definedExternally

@kotlin.internal.InlineOnly
public inline fun MediaTrackSettings.aspectRatio.set(value) = definedExternally

@kotlin.internal.InlineOnly
public inline fun MediaTrackSettings.frameRate.set(value) = definedExternally

@kotlin.internal.InlineOnly
public inline fun MediaTrackSettings.facingMode.set(value) = definedExternally

@kotlin.internal.InlineOnly
public inline fun MediaTrackSettings.resizeMode.set(value) = definedExternally

@kotlin.internal.InlineOnly
public inline fun MediaTrackSettings.deviceId.set(value) = definedExternally

@kotlin.internal.InlineOnly
public inline fun MediaTrackSettings.groupId.set(value) = definedExternally

@kotlin.internal.InlineOnly
public inline fun MediaTrackSettings.volume.set(value) = definedExternally

@kotlin.internal.InlineOnly
public inline fun MediaTrackSettings.sampleRate.set(value) = definedExternally

@kotlin.internal.InlineOnly
public inline fun MediaTrackSettings.sampleSize.set(value) = definedExternally

@kotlin.internal.InlineOnly
public inline fun MediaTrackSettings.echoCancellation.set(value) = definedExternally

@kotlin.internal.InlineOnly
public inline fun MediaTrackSettings.autoGainControl.set(value) = definedExternally

@kotlin.internal.InlineOnly
public inline fun MediaTrackSettings.noiseSuppression.set(value) = definedExternally

@kotlin.internal.InlineOnly
public inline fun MediaTrackSettings.latency.set(value) = definedExternally

@kotlin.internal.InlineOnly
public inline fun MediaTrackSettings.channelCount.set(value) = definedExternally

Open Source Used in security Automation Identity-guest-cs 1718
definedExternally set(value) = definedExternally
var volume: Double? = undefined
get() = undefined

set(value) = definedExternally
var sampleRate: Int? = undefined
get() = undefined

set(value) = definedExternally
var sampleSize: Int? = undefined
get() = undefined

set(value) = definedExternally
var echoCancellation: Boolean? = undefined
get() = undefined

set(value) = definedExternally
var autoGainControl: Boolean? = undefined
get() = undefined

set(value) = definedExternally
var noiseSuppression: Boolean? = undefined
get() = undefined

set(value) = definedExternally
var latency: Double? = undefined
get() = undefined

set(value) = definedExternally
var channelCount: Int? = undefined
get() = undefined

set(value) = definedExternally
var deviceId: String? = undefined
get() = undefined

set(value) = definedExternally
var groupId: String? = undefined
get() = undefined

set(value) = definedExternally
    val o = js("({})")
    o["width"] = width
    o["height"] = height
    o["aspectRatio"] = aspectRatio
    o["frameRate"] = frameRate
    o["facingMode"] = facingMode
    o["resizeMode"] = resizeMode
    o["volume"] = volume
    o["sampleRate"] = sampleRate
    o["sampleSize"] = sampleSize
    o["echoCancellation"] = echoCancellation
    o["autoGainControl"] = autoGainControl
    o["noiseSuppression"] = noiseSuppression
    o["latency"] = latency
    o["channelCount"] = channelCount
    o["deviceId"] = deviceId
    o["groupId"] = groupId
    return o
}

/**
 * Exposes the JavaScript [MediaStreamTrackEvent](https://developer.mozilla.org/en/docs/Web/API/MediaStreamTrackEvent) to Kotlin public open class MediaStreamTrackEvent(type: String, eventInitDict: MediaStreamTrackEventInit) : Event {
    open var track: MediaStreamTrack?

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public interface MediaStreamTrackEventInit : EventInit {
    var track: MediaStreamTrack?
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["track"] = track
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}

/**
 * Exposes the JavaScript [MediaDevices](https://developer.mozilla.org/en/docs/Web/API/MediaDevices) to Kotlin public abstract class MediaDevices : EventTarget {
    open var deviceChange: ((Event) -> dynamic)?

    fun enumerateDevices(): Promise<Array<MediaDeviceInfo>>
    fun getSupportedConstraints(): MediaTrackSupportedConstraints
    fun getUserMedia(constraints: MediaStreamConstraints): MediaStream
}

/**
 * Exposes the JavaScript [MediaDeviceInfo](https://developer.mozilla.org/en/docs/Web/API/MediaDeviceInfo) to Kotlin public abstract class MediaDeviceInfo {
    open var label: String
        get() = undefined
        set(value) = undefined
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["label"] = label
    o["vendor"] = vendor
    o["product"] = product
    o["serialNumber"] = serialNumber
    o["manufacturer"] = manufacturer
    o["compatibleProducts"] = compatibleProducts
    return o
}
JavaScript [MediaStreamConstraints](https://developer.mozilla.org/en/docs/Web/API/MediaStreamConstraints) to Kotlin

```kotlin
public external interface MediaStreamConstraints {
    var video: dynamic /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var audio: dynamic /* = false */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun MediaStreamConstraints(video: dynamic = false, audio: dynamic = false): MediaStreamConstraints {
    return o
}

@kotlin.internal.InlineOnly
public inline fun DoubleRange(max: Double? = undefined, min: Double? = undefined): DoubleRange {
    return o
}
```

---

Open Source Used in security_automation identity-guest-cs 1720
```kotlin
var advanced: Array<ConstraintSet>? = null
get() = definedExternally
set(value) = definedExternally

public external fun Constraints(advanced: Array<ConstraintSet>? = undefined): Constraints

/* please, don't implement this interface! */
public external interface MediaStreamTrackState {
    companion object
}

public inline val MediaStreamTrackState.Companion.LIVE: MediaStreamTrackState
    get() = "live".asDynamic().unsafeCast<MediaStreamTrackState>()

public inline val MediaStreamTrackState.Companion.ENDED: MediaStreamTrackState
    get() = "ended".asDynamic().unsafeCast<MediaStreamTrackState>()

/* please, don't implement this interface! */
public external interface VideoFacingModeEnum {
    companion object
}

public inline val VideoFacingModeEnum.Companion.USER: VideoFacingModeEnum
    get() = "user".asDynamic().unsafeCast<VideoFacingModeEnum>()

public inline val VideoFacingModeEnum.Companion.ENVIRONMENT: VideoFacingModeEnum
    get() = "environment".asDynamic().unsafeCast<VideoFacingModeEnum>()

/* please, don't implement this interface! */
public external interface VideoResizeModeEnum {
    companion object
}

public inline val VideoResizeModeEnum.Companion.NONE: VideoResizeModeEnum
    get() = "none".asDynamic().unsafeCast<VideoResizeModeEnum>()

public inline val VideoResizeModeEnum.Companion.CROP_AND_SCALE: VideoResizeModeEnum
    get() = "crop-and-scale".asDynamic().unsafeCast<VideoResizeModeEnum>()

/* please, don't implement this interface! */
public external interface MediaDeviceKind {
    companion object
}

public inline val MediaDeviceKind.Companion.AUDIOINPUT: MediaDeviceKind
    get() = "audioinput".asDynamic().unsafeCast<MediaDeviceKind>()

public inline val MediaDeviceKind.Companion.AUDIOOUTPUT: MediaDeviceKind
    get() = "audiooutput".asDynamic().unsafeCast<MediaDeviceKind>()

public inline val MediaDeviceKind.Companion.VIDEOINPUT: MediaDeviceKind
    get() = "videoinput".asDynamic().unsafeCast<MediaDeviceKind>()
```

---

**Open Source Used In security_automation identity-guest-cs**

[1721]
    val o = js("({})")
    o["pointerId"] = pointerId
    o["width"] = width
    o["height"] = height
    o["pressure"] = pressure
    o["tangentialPressure"] = tangentialPressure
    o["tiltX"] = tiltX
    o["tiltY"] = tiltY
    o["twist"] = twist
    o["pointerType"] = pointerType
    o["isPrimary"] = isPrimary
    o["screenX"] = screenX
    o["screenY"] = screenY
    o["clientX"] = clientX
    o["clientY"] = clientY
    o["button"] = button
    o["buttons"] = buttons
    o["relatedTarget"] = relatedTarget
    o["region"] = region
    o["ctrlKey"] = ctrlKey
    o["shiftKey"] = shiftKey
    o["altKey"] = altKey
    o["metaKey"] = metaKey
    o["modifierAltGraph"] = modifierAltGraph
    o["modifierCapsLock"] = modifierCapsLock
    o["modifierFn"] = modifierFn
    o["modifierFnLock"] = modifierFnLock
    o["modifierHyper"] = modifierHyper
    o["modifierNumLock"] = modifierNumLock
    o["modifierScrollLock"] = modifierScrollLock
    o["modifierSuper"] = modifierSuper
    o["modifierSymbol"] = modifierSymbol
    o["modifierSymbolLock"] = modifierSymbolLock
    o["view"] = view
    o["detail"] = detail
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}

/**
 * Exposes the JavaScript [PointerEvent](https://developer.mozilla.org/en/docs/Web/API/PointerEvent) to Kotlin
 */
public external open class PointerEvent(type: String, eventInitDict: PointerEventInit = definedExternally) : MouseEvent {
    open val pointerId: Int
    open val width: Double
    open val height: Double
    open val pressure: Float
    open val tangentialPressure: Float
    open val tiltX: Int
    open val tiltY: Int
    open val twist: Int
    open val pointerType: String
    open val isPrimary: Boolean

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}
definedExternally
var clipped: Boolean? /* = false */
get() = definedExternally
set(value) =
definedExternally

@kotlin.internal.InlineOnly
val o = js("({})")
    o["fill"] = fill
    o["stroke"] = stroke
    o["markers"] = markers
    o["clipped"] = clipped
    return o
}

/**
 * Exposes the JavaScript

 * Exposes the JavaScript
 */
public external abstract class SVGGraphicsElement : SVGElement, SVGTests {
    open val transform: SVGAnimatedTransformList
    fun getBBox(options: SVGBoundingBoxOptions = definedExternally): DOMRect
    fun getCTM(): DOMMatrix
    fun getScreenCTM(): DOMMatrix

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript
 */
public external abstract class SVGGeometryElement : SVGGraphicsElement {
    open val pathLength: SVGAnimatedNumber
    fun isPointInFill(point: DOMPoint): Boolean
    fun isPointInStroke(point: DOMPoint): Boolean
    fun getTotalLength(): Float
    fun getPointAtLength(distance: Float): DOMPoint

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript
 */
public external abstract class SVGNumber {
    open var value: Float
}

/**
 * Exposes the JavaScript
 */
public external abstract class SVGLength {
    open val unitType: Short
    open var value: Float
    open var valueInSpecifiedUnits: Float
    open var valueAsString: String
    fun newValueSpecifiedUnits(unitType: Short, valueInSpecifiedUnits: Float)
    fun convertToSpecifiedUnits(unitType: Short)

    companion object {
        val SVG_LENGTHTYPE_UNKNOWN: Short
        val SVG_LENGTHTYPE_NUMBER: Short
        val SVG_LENGTHTYPE_PERCENTAGE: Short
        val SVG_LENGTHTYPE_EXS: Short
        val SVG_LENGTHTYPE_CM: Short
        val SVG_LENGTHTYPE_MM: Short
        val SVG_LENGTHTYPE_IN: Short
        val SVG_LENGTHTYPE_PT: Short
        val SVG_LENGTHTYPE_PC: Short
    }
}

/**
 * Exposes the JavaScript
 */
public external abstract class SVGAngle {
    open val unitType: Short
    open var value: Float
    open var valueInSpecifiedUnits: Float
    open var valueAsString: String
    fun newValueSpecifiedUnits(unitType: Short, valueInSpecifiedUnits: Float)
    fun convertToSpecifiedUnits(unitType: Short)

    companion object {
        val SVG_ANGLETYPE_UNKNOWN: Short
        val SVG_ANGLETYPE_UNSPECIFIED: Short
        val
```kotlin
public external abstract class SVGNameList {
    open val length: Int
    open val numberOfItems: Int
    fun clear()
    fun initialize(newItem: dynamic): dynamic
    fun insertItemBefore(newItem: dynamic, index: Int): dynamic
    fun replaceItem(newItem: dynamic, index: Int): dynamic
    fun removeItem(index: Int): dynamic
    fun appendItem(newItem: dynamic): dynamic
    fun getItem(index: Int): dynamic
}

@kotlin.internal.InlineOnly
public inline operator fun SVGNameList.get(index: Int): dynamic = asDynamic()[index]

@kotlin.internal.InlineOnly
public inline operator fun SVGNameList.set(index: Int, newItem: dynamic) { asDynamic()[index] = newItem }

/**
 * Exposes the JavaScript [SVGNumberList](https://developer.mozilla.org/en/docs/Web/API/SVGNumberList) to Kotlin
 */
public external abstract class SVGNumberList {
    open val length: Int
    open val numberOfItems: Int
    fun clear()
    fun initialize(newItem: SVGNumber): SVGNumber
    fun insertItemBefore(newItem: SVGNumber, index: Int): SVGNumber
    fun replaceItem(newItem: SVGNumber, index: Int): SVGNumber
    fun removeItem(index: Int): SVGNumber
    fun appendItem(newItem: SVGNumber): SVGNumber
    fun getItem(index: Int): SVGNumber
}

@kotlin.internal.InlineOnly
public inline operator fun SVGNumberList.get(index: Int): SVGNumber? = asDynamic()[index]

@kotlin.internal.InlineOnly
public inline operator fun SVGNumberList.set(index: Int, newItem: SVGNumber) { asDynamic()[index] = newItem }

/**
 * Exposes the JavaScript [SVGLengthList](https://developer.mozilla.org/en/docs/Web/API/SVGLengthList) to Kotlin
 */
public external abstract class SVGLengthList {
    open val length: Int
    open val numberOfItems: Int
    fun clear()
    fun initialize(newItem: SVGLength): SVGLength
    fun insertItemBefore(newItem: SVGLength, index: Int): SVGLength
    fun replaceItem(newItem: SVGLength, index: Int): SVGLength
    fun removeItem(index: Int): SVGLength
    fun appendItem(newItem: SVGLength): SVGLength
    fun getItem(index: Int): SVGLength
}

@kotlin.internal.InlineOnly
public inline operator fun SVGLengthList.get(index: Int): SVGLength? = asDynamic()[index]

@kotlin.internal.InlineOnly
public inline operator fun SVGLengthList.set(index: Int, newItem: SVGLength) { asDynamic()[index] = newItem }

/**
 * Exposes the JavaScript [SVGAnimatedBoolean](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedBoolean) to Kotlin
 */
public external abstract class SVGAnimatedBoolean {
    open var baseVal: Boolean
    open val animVal: Boolean
}

/**
 * Exposes the JavaScript [SVGAnimatedEnumeration](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedEnumeration) to Kotlin
 */
public external abstract class SVGAnimatedEnumeration {
    open var baseVal: Short
    open val animVal: Short
}

/**
 * Exposes the JavaScript [SVGAnimatedInteger](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedInteger) to Kotlin
 */
public external abstract class SVGAnimatedInteger {
    open var baseVal: Int
    open val animVal: Int
}

/**
 * Exposes the JavaScript [SVGAnimatedNumber](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedNumber) to Kotlin
 */
public external abstract class SVGAnimatedNumber {
    open var baseVal: Float
    open val animVal: Float
}

/**
 * Exposes the JavaScript [SVGAnimatedLength](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedLength) to Kotlin
 */
public external abstract class SVGAnimatedLength {
    open val baseVal: SVGLength
    open val animVal: SVGLength
}

/**
 * Exposes the JavaScript [SVGAnimatedAngle](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedAngle) to Kotlin
 */
public val SVGAnimatedAngle
```
public abstract class SVGAnimatedNumberList {
    open val baseVal: SVGNumberList
    open val animVal: SVGNumberList
}

/**
* Exposes the JavaScript
*/
public abstract class SVGAnimatedLengthList {
    open val baseVal: SVGLengthList
    open val animVal: SVGLengthList
}

/**
* Exposes the JavaScript
*/
public abstract class SVGStringList {
    open val length: Int
    open val numberOfItems: Int
    fun clear()
    fun insertItemBefore(newItem: String, index: Int): String
    fun replaceItem(newItem: String, index: Int): String
    fun appendItem(newItem: String): String
    fun getItems(index: Int): String
    fun set(index: Int, newItem: String): String
}

@kotlin.internal.InlineOnly
public inline operator fun SVGStringList.get(index: Int): String? = asDynamic()[index]

@kotlin.internal.InlineOnly
public inline operator fun SVGStringList.set(index: Int, newItem: String) { asDynamic()[index] = newItem }

/**
* Exposes the JavaScript
*/
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public interface SVGUnitTypes {
    companion object {
        val SVG_UNIT_TYPE_UNKNOWN: Short
        val SVG_UNIT_TYPE_USERSPACEONUSE: Short
        val SVG_UNIT_TYPE_OBJECTBOUNDINGBOX: Short
    }
}

/**
* Exposes the JavaScript
*/
public interface SVGTests {
    val requiredExtensions: SVGStringList
    val systemLanguage: SVGStringList
}

public interface SVGFitToViewBox {
    val viewBox: SVGAnimatedRect
    val preserveAspectRatio: SVGAnimatedPreserveAspectRatio
}

/**
* Exposes the JavaScript
*/
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public interface SVGZoomAndPan {
    var zoomAndPan: Short

    companion object {
        val SVG_ZOOMANDPAN_UNKNOWN: Short
        val SVG_ZOOMANDPAN_DISABLE: Short
        val SVG_ZOOMANDPAN_MAGNIFY: Short
    }
}

/**
* Exposes the JavaScript
*/
public interface SVGURIReference {
    val href: SVGAnimatedString
}

/**
* Exposes the JavaScript
*/
public abstract class SVGSVGElement : SVGGraphicsElement, SVGFitToViewBox, SVGZoomAndPan, WindowEventHandlers {
    open val x: SVGAnimatedLength
    open val y: SVGAnimatedLength
    open val width: SVGAnimatedLength
    open val height: SVGAnimatedLength
    open var currentScale: Float
    open var currentTranslate: DOMPointReadOnly
    fun getIntersectionList(rect: DOMRectReadOnly, referenceElement: SVGElement?): NodeList
    fun getEnclosureList(rect: DOMRectReadOnly, referenceElement: SVGElement?): NodeList
    fun checkIntersection(element: SVGElement, rect: DOMRectReadOnly): Boolean
    fun checkEnclosure(element: SVGElement, rect: DOMRectReadOnly): Boolean
    fun deselectAll()
    fun createSVGNumber(): SVGNumber
    fun createSVGLength(): SVGLength
    fun createSVGAngle(): SVGAngle
    fun createSVGPoint(): DOMPoint
    fun createSVGMatrix(): DOMMatrix
    fun createSVGRect(): DOMRect
    fun createSVGTransform(): SVGTransform
    fun createSVGTransformFromMatrix(matrix: DOMMatrix): SVGTransform
    fun getElementById(elementId: String): Element
    fun suspendRedraw(maxWaitMilliseconds: Int): Int
    fun unsuspendRedraw(suspendHandleID: Int): Int
    fun forceRedraw()

    companion object {
        val SVG_ZOOMANDPAN_UNKNOWN: Short
        val SVG_ZOOMANDPAN_DISABLE: Short
        val SVG_ZOOMANDPAN_MAGNIFY: Short
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val
/**
 * Exposes the JavaScript [SVGMetadataElement](https://developer.mozilla.org/en/docs/Web/API/SVGMetadataElement) to Kotlin
 */
public external abstract class SVGMetadataElement : SVGElement {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [SVGTitleElement](https://developer.mozilla.org/en/docs/Web/API/SVGTitleElement) to Kotlin
 */
public external abstract class SVGTitleElement : SVGElement {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [SVGSymbolElement](https://developer.mozilla.org/en/docs/Web/API/SVGSymbolElement) to Kotlin
 */
public external abstract class SVGSymbolElement : SVGGraphicsElement, SVGFitToViewBox {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [SVGUseElement](https://developer.mozilla.org/en/docs/Web/API/SVGUseElement) to Kotlin
 */
public external abstract class SVGUseElement : SVGGraphicsElement, SVGURIReference {
    open val x: SVGAnimatedLength
    open val y: SVGAnimatedLength
    open val width: SVGAnimatedLength
    open val height: SVGAnimatedLength
    open val instanceRoot: SVGElement?
    open val animatedInstanceRoot: SVGElement?

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

public external open class SVGUseElementShadowRoot : ShadowRoot {
    companion object {
        val ELEMENT_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}

public external interface SVGUseElementShadowRoot : ShadowRoot {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}

public external open class SVGElementInstance {
    val correspondingElement: SVGElement?
        get() = definedExternally
    val correspondingUseElement: SVGUseElement?
        get() = definedExternally
}

public external open class ShadowAnimation(source: dynamic, newTarget: dynamic) {
    open val sourceAnimation: dynamic
}

/**
 * Exposes the JavaScript [SVGSwitchElement](https://developer.mozilla.org/en/docs/Web/API/SVGSwitchElement) to Kotlin
 */
public external abstract class SVGSwitchElement : SVGGraphicsElement {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}

public external interface GetSVGDocument {
    fun getSVGDocument(): Document
}

/**
 * Exposes the JavaScript [SVGStyleElement](https://developer.mozilla.org/en/docs/Web/API/SVGStyleElement) to Kotlin
 */
public external abstract class SVGStyleElement : SVGElement, LinkStyle {
    open var type: String
    open var media: String
    open var title: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}

/**
 * Exposes the JavaScript [SVGTransform](https://developer.mozilla.org/en/docs/Web/API/SVGTransform) to Kotlin
 */
public external abstract class SVGTransform {
    open val type: Short
    open val matrix: DOMMatrix
    open val angle: Float
    fun setMatrix(matrix: DOMMatrixReadOnly)
    fun setTranslate(tx: Float, ty: Float)
    fun setScale(sx: Float, sy: Float)
    fun setRotate(angle: Float, cx: Float, cy: Float)
    fun setSkewX(angle: Float)
    fun setSkewY(angle: Float)

    companion object {
        val SVG_TRANSFORM_UNKNOWN: Short
        val SVG_TRANSFORM_MATRIX: Short
        val SVG_TRANSFORM_TRANSLATE: Short
        val SVG_TRANSFORM_SCALE: Short
        val SVG_TRANSFORM_ROTATE: Short
        val SVG_TRANSFORM_SKEWX: Short
        val SVG_TRANSFORM_SKEWY: Short
    }
}

/**
 * Exposes the JavaScript [SVGTransformList](https://developer.mozilla.org/en/docs/Web/API/SVGTransformList) to Kotlin
 */
public external abstract class SVGTransformList {
    open val length: Int
    open val numberOfItems: Int

fun clear():
    fun initialize(newItem: SVGTransform):
    fun insertItemBefore(newItem: SVGTransform, index: Int):
    fun replaceItem(newItem: SVGTransform, index: Int):
    fun createSVGTransformFromMatrix(matrix: DOMMatrixReadOnly):
    fun consolidate():
    fun getItem(index: Int):
}

@kotlin.internal.InlineOnly
public inline operator fun SVGTransformList.get(index: Int):

@kotlin.internal.InlineOnly
public inline operator fun SVGTransformList.set(index: Int, newItem: SVGTransform):
/**
 * Exposes the JavaScript [SVGCircleElement](https://developer.mozilla.org/en/docs/Web/API/SVGCircleElement) to Kotlin
 */
public external abstract class SVGCircleElement : SVGGeometryElement {
    open val cx: SVGAnimatedLength
    open val cy: SVGAnimatedLength
    open val r: SVGAnimatedLength

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [SVGEllipseElement](https://developer.mozilla.org/en/docs/Web/API/SVGEllipseElement) to Kotlin
 */
public external abstract class SVGEllipseElement : SVGGeometryElement {
    open val cx: SVGAnimatedLength
    open val cy: SVGAnimatedLength
    open val rx: SVGAnimatedLength
    open val ry: SVGAnimatedLength

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [SVGLineElement](https://developer.mozilla.org/en/docs/Web/API/SVGLineElement) to Kotlin
 */
public external abstract class SVGLineElement : SVGGeometryElement {
    open val x1: SVGAnimatedLength
    open val y1: SVGAnimatedLength
    open val x2: SVGAnimatedLength
    open val y2: SVGAnimatedLength

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [SVGMeshElement](https://developer.mozilla.org/en/docs/Web/API/SVGMeshElement) to Kotlin
 */
public external abstract class SVGMeshElement : SVGGeometryElement, SVGURIReference {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
**Exposes the JavaScript [SVGAnimatedPoints](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedPoints) to Kotlin**

```kotlin
public external interface SVGAnimatedPoints {
    val points: SVGPointList
    val animatedPoints: SVGPointList
}

public external abstract class SVGPointList {
    open val length: Int
    open val numberOfItems: Int
    fun clear()
    fun initialize(newItem: DOMPoint): DOMPoint
    fun insertItemBefore(newItem: DOMPoint, index: Int): DOMPoint
    fun replaceItem(newItem: DOMPoint, index: Int): DOMPoint
    fun removeItem(index: Int): DOMPoint
    fun appendItem(newItem: DOMPoint): DOMPoint
    fun getItem(index: Int): DOMPoint
}

@kotlin.internal.InlineOnly
public inline operator fun SVGPointList.get(index: Int): DOMPoint? = asDynamic()[index]

@kotlin.internal.InlineOnly
public inline operator fun SVGPointList.set(index: Int, newItem: DOMPoint) { asDynamic()[index] = newItem }

**Exposes the JavaScript [SVGPolylineElement](https://developer.mozilla.org/en/docs/Web/API/SVGPolylineElement) to Kotlin**

```kotlin
public external abstract class SVGPolylineElement : SVGGeometryElement, SVGAnimatedPoints {
}
```

**Exposes the JavaScript [SVGPolygonElement](https://developer.mozilla.org/en/docs/Web/API/SVGPolygonElement) to Kotlin**

```kotlin
public external abstract class SVGPolygonElement : SVGGeometryElement, SVGAnimatedPoints {
}
```

**Exposes the JavaScript [SVGTextContentElement](https://developer.mozilla.org/en/docs/Web/API/SVGPolylineElement) to Kotlin**

```kotlin
public external abstract class SVGTextContentElement : SVGGraphicsElement {
    open val textLength: SVGAnimatedLength
    open val lengthAdjust: SVGAnimatedEnumeration
    fun getNumberOfChars(): Int
    fun getComputedTextLength(): Float
    fun getSubStringLength(charnum: Int, nchars: Int): Float
    fun getStartPositionOfChar(charnum: Int): DOMPoint
    fun getEndPositionOfChar(charnum: Int): DOMPoint
    fun getExtentOfChar(charnum: Int): DOMRect
    fun getRotationOfChar(charnum: Int): Float
    fun getCharNumAtPosition(point: DOMPoint): Int
    fun selectSubString(charnum: Int, nchars: Int)

    companion object {
        val LENGTHADJUST_UNKNOWN: Short
        val LENGTHADJUST_SPACING: Short
        val LENGTHADJUST_SPACINGANDGLYPHS: Short
    }
}
```
**Exposes the JavaScript [SVGTextPositioningElement](https://developer.mozilla.org/en/docs/Web/API/SVGTextPositioningElement) to Kotlin
 */
public external abstract class SVGTextPositioningElement : SVGTextContentElement {
    open val x: SVGAnimatedLengthList
    open val y: SVGAnimatedLengthList
    open val dx: SVGAnimatedLengthList
    open val dy: SVGAnimatedLengthList
    open val rotate: SVGAnimatedNumberList

    companion object {
        val LENGTHADJUST_UNKNOWN: Short
        val LENGTHADJUST_SPACING: Short
        val LENGTHADJUST_SPACINGANDGLYPHS: Short
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

**Exposes the JavaScript [SVGTextElement](https://developer.mozilla.org/en/docs/Web/API/SVGTextElement) to Kotlin
 */
public external abstract class SVGTextElement : SVGTextPositioningElement {
    companion object {
        val LENGTHADJUST_UNKNOWN: Short
        val LENGTHADJUST_SPACING: Short
        val LENGTHADJUST_SPACINGANDGLYPHS: Short
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

**Exposes the JavaScript [SVGTSpanElement](https://developer.mozilla.org/en/docs/Web/API/SVGTSpanElement) to Kotlin
 */
public external abstract class SVGTSpanElement : SVGTextPositioningElement {
    companion object {
        val LENGTHADJUST_UNKNOWN: Short
        val LENGTHADJUST_SPACING: Short
        val LENGTHADJUST_SPACINGANDGLYPHS: Short
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }

**Exposes the JavaScript [SVGTextPathElement](https://developer.mozilla.org/en/docs/Web/API/SVGTextPathElement) to Kotlin
 */
public external abstract class SVGTextPathElement : SVGTextContentElement, SVGURIReference {
    open val startOffset: SVGAnimatedLength
    open val method: SVGAnimatedEnumeration
    open val spacing: SVGAnimatedEnumeration
    companion object {
        val TEXTPATH_METHODTYPE_UNKNOWN: Short
        val TEXTPATH_METHODTYPE_STRETCH: Short
        val TEXTPATH_METHODTYPE_SPACINGTYPE_UNKNOWN: Short
        val TEXTPATH_SPACINGTYPE_AUTO: Short
        val TEXTPATH_SPACINGTYPE_EXACT: Short
        val TEXTPATH_SPACINGANDGLYPHS: Short
        val ELEMENT_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ATTRIBUTE_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        open val TEXT_PATH_TYPE_UNKNOWN: SVGAnimatedEnumeration
        open val TEXT_PATH_METHODTYPE_ALIGN: SVGAnimatedEnumeration
        open val TEXTPATH_SPACINGTYPE_AUTO: SVGAnimatedEnumeration
        open val TEXTPATH_SPACINGTYPE_EXACT: SVGAnimatedEnumeration
        open val LENGTHADJUST_UNKNOWN: SVGAnimatedEnumeration
        open val LENGTHADJUST_SPACING: SVGAnimatedEnumeration
        open val LENGTHADJUST_SPACINGANDGLYPHS: SVGAnimatedEnumeration
        open val ATTRIBUT

fun setOrientToAngle(angle: SVGAngle)

companion object {
    val SVG_MARKERUNITS_UNKNOWN: Short
    val SVG_MARKERUNITS_USERSPACEONUSE: Short
    val SVG_MARKERUNITS_STROKEWIDTH: Short
    val SVG_MARKER_ORIENT_UNKNOWN: Short
    val SVG_MARKER_ORIENT_AUTO: Short
    val SVG_MARKER_ORIENT_ANGLE: Short
    val ELEMENT_NODE: Short
    val ATTRIBUTE_NODE: Short
    val TEXT_NODE: Short
    val CDATA_SECTION_NODE: Short
    val ENTITY_REFERENCE_NODE: Short
    val ENTITY_NODE: Short
    val PROCESSING_INSTRUCTION_NODE: Short
    val COMMENT_NODE: Short
    val DOCUMENT_NODE: Short
    val DOCUMENT_TYPE_NODE: Short
    val DOCUMENT_FRAGMENT_NODE: Short
    val NOTATION_NODE: Short
    val DOCUMENT_POSITION_DISCONNECTED: Short
    val DOCUMENT_POSITION_PRECEDING: Short
    val DOCUMENT_POSITION_FOLLOWING: Short
    val DOCUMENT_POSITION_CONTAINS: Short
    val DOCUMENT_POSITION_CONTAINED_BY: Short
    val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
}

/**
 * Exposes the JavaScript [SVGSolidcolorElement](https://developer.mozilla.org/en/docs/Web/API/SVGSolidcolorElement) to Kotlin
 */
public external abstract class SVGSolidcolorElement : SVGElement {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [SVGGradientElement](https://developer.mozilla.org/en/docs/Web/API/SVGGradientElement) to Kotlin
 */
public external abstract class SVGGradientElement : SVGElement, SVGURIReference, SVGUnitTypes {
    open val gradientUnits: SVGAnimatedEnumeration
    open val gradientTransform: SVGAnimatedTransformList
    open val spreadMethod: SVGAnimatedEnumeration
    companion object {
        val SVG_SPREADMETHOD_UNKNOWN: Short
        val SVG_SPREADMETHOD_PAD: Short
        val SVG_SPREADMETHOD_REFLECT: Short
        val SVG_SPREADMETHOD_REPEAT: Short
        val SVG_UNIT_TYPE_UNKNOWN: Short
        val SVG_UNIT_TYPE_USERSPACEONUSE: Short
        val SVG_UNIT_TYPE_OBJECTBOUNDINGBOX: Short
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }

/**
 * Exposes the JavaScript [SVGLinearGradientElement](https://developer.mozilla.org/en/docs/Web/API/SVGLinearGradientElement) to Kotlin
 */
public external abstract class SVGLinearGradientElement : SVGGradientElement {
    open val x1: SVGAnimatedLength
    open val y1: SVGAnimatedLength
    open val x2: SVGAnimatedLength
    open val y2: SVGAnimatedLength
    companion object {
        val SVG_SPREADMETHOD_UNKNOWN: Short
        val SVG_SPREADMETHOD_PAD: Short
        val SVG_SPREADMETHOD_REFLECT: Short
        val SVG_SPREADMETHOD_REPEAT: Short
        val SVG_UNIT_TYPE_UNKNOWN: Short
        val SVG_UNIT_TYPE_USERSPACEONUSE: Short
        val SVG_UNIT_TYPE_OBJECTBOUNDINGBOX: Short
    }
}
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

SVGMeshpatchElement : SVGElement {
  companion object {
    val ELEMENT_NODE: Short
    val ATTRIBUTE_NODE: Short
    val TEXT_NODE: Short
    val CDATA_SECTION_NODE: Short
    val ENTITY_REFERENCE_NODE: Short
    val ENTITY_NODE: Short
    val PROCESSING_INSTRUCTION_NODE: Short
    val COMMENT_NODE: Short
    val DOCUMENT_NODE: Short
    val DOCUMENT_TYPE_NODE: Short
    val DOCUMENT_FRAGMENT_NODE: Short
    val NOTATION_NODE: Short
    val DOCUMENT_POSITION_DISCONNECTED: Short
    val DOCUMENT_POSITION_PRECEDING: Short
    val DOCUMENT_POSITION_FOLLOWING: Short
    val DOCUMENT_POSITION_CONTAINS: Short
    val DOCUMENT_POSITION_CONTAINED_BY: Short
    val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
  }
}

public external abstract class SVGMeshpatchElement : SVGElement {
  companion object {
    val ELEMENT_NODE: Short
    val ATTRIBUTE_NODE: Short
    val TEXT_NODE: Short
    val CDATA_SECTION_NODE: Short
    val ENTITY_REFERENCE_NODE: Short
    val ENTITY_NODE: Short
    val PROCESSING_INSTRUCTION_NODE: Short
    val COMMENT_NODE: Short
    val DOCUMENT_NODE: Short
    val DOCUMENT_TYPE_NODE: Short
    val DOCUMENT_FRAGMENT_NODE: Short
    val NOTATION_NODE: Short
    val DOCUMENT_POSITION_DISCONNECTED: Short
    val DOCUMENT_POSITION_PRECEDING: Short
    val DOCUMENT_POSITION_FOLLOWING: Short
    val DOCUMENT_POSITION_CONTAINS: Short
    val DOCUMENT_POSITION_CONTAINED_BY: Short
    val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
  }
}

/**
 * Exposes the JavaScript [SVGStopElement](https://developer.mozilla.org/en/docs/Web/API/SVGStopElement) to Kotlin
 */
public external abstract class SVGStopElement : SVGElement {
  open val offset: SVGAnimatedNumber

  companion object {
    val ELEMENT_NODE: Short
    val ATTRIBUTE_NODE: Short
    val TEXT_NODE: Short
    val CDATA_SECTION_NODE: Short
    val ENTITY_REFERENCE_NODE: Short
    val ENTITY_NODE: Short
    val PROCESSING_INSTRUCTION_NODE: Short
    val COMMENT_NODE: Short
    val DOCUMENT_NODE: Short
    val DOCUMENT_TYPE_NODE: Short
    val DOCUMENT_FRAGMENT_NODE: Short
    val NOTATION_NODE: Short
    val DOCUMENT_POSITION_DISCONNECTED: Short
    val DOCUMENT_POSITION_PRECEDING: Short
    val DOCUMENT_POSITION_FOLLOWING: Short
    val DOCUMENT_POSITION_CONTAINS: Short
    val DOCUMENT_POSITION_CONTAINED_BY: Short
    val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
  }
}

/**
 * Exposes the JavaScript [SVGPatternElement](https://developer.mozilla.org/en/docs/Web/API/SVGPatternElement) to Kotlin
 */
public external abstract class SVGPatternElement : SVGElement, SVGFitToViewBox, SVGURIReference, SVGUnitTypes {
  open val patternUnits: SVGAnimatedEnumeration
  open val patternContentUnits: SVGAnimatedEnumeration
  open val patternTransform: SVGAnimatedTransformList
  open val x: SVGAnimatedLength
  open val y: SVGAnimatedLength
  open val width: SVGAnimatedLength
  open val height: SVGAnimatedLength

  companion object {
    val SVG_UNIT_TYPE_UNKNOWN: Short
    val SVG_UNIT_TYPE_USERSPACEONUSE: Short
    val SVG_UNIT_TYPE_OBJECTBOUNDINGBOX: Short
    val ELEMENT_NODE: Short
    val ATTRIBUTE_NODE: Short
    val TEXT_NODE: Short
    val CDATA_SECTION_NODE: Short
    val ENTITY_REFERENCE_NODE: Short
    val ENTITY_NODE: Short
    val PROCESSING_INSTRUCTION_NODE: Short
    val COMMENT_NODE: Short
    val DOCUMENT_NODE: Short
    val DOCUMENT_TYPE_NODE: Short
    val DOCUMENT_FRAGMENT_NODE: Short
    val NOTATION_NODE: Short
    val DOCUMENT_POSITION_DISCONNECTED: Short
    val DOCUMENT_POSITION_PRECEDING: Short
    val DOCUMENT_POSITION_FOLLOWING: Short
    val DOCUMENT_POSITION_CONTAINS: Short
    val DOCUMENT_POSITION_CONTAINED_BY: Short
    val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
  }
}

public external abstract class SVGHatchElement : SVGElement {
  companion object {
    val ELEMENT_NODE: Short
    val ATTRIBUTE_NODE: Short
    val TEXT_NODE: Short
    val CDATA_SECTION_NODE: Short
    val ENTITY_REFERENCE_NODE: Short
    val ENTITY_NODE: Short
    val PROCESSING_INSTRUCTION_NODE: Short
    val COMMENT_NODE: Short
    val DOCUMENT_NODE: Short
    val DOCUMENT_TYPE_NODE: Short
    val DOCUMENT_FRAGMENT_NODE: Short
    val NOTATION_NODE: Short
    val DOCUMENT_POSITION_DISCONNECTED: Short
    val DOCUMENT_POSITION_PRECEDING: Short
    val DOCUMENT_POSITION_FOLLOWING: Short
    val DOCUMENT_POSITION_CONTAINS: Short
    val DOCUMENT_POSITION_CONTAINED_BY: Short
    val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
  }
}
public external abstract class SVGPathElement : SVGElement {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [SVGPathElement](https://developer.mozilla.org/en/docs/Web/API/SVGPathElement) to Kotlin
 */
public external abstract class SVGPathElement : SVGElement, SVGURIReference {
    open val x: SVGAnimatedLength
    open val y: SVGAnimatedLength

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [SVGRectElement](https://developer.mozilla.org/en/docs/Web/API/SVGRectElement) to Kotlin
 */
public external abstract class SVGRectElement : SVGElement, SVGURIReference {
    open val x: SVGAnimatedLength
    open val y: SVGAnimatedLength
    open val width: SVGAnimatedLength
    open val height: SVGAnimatedLength

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [SVGTextElement](https://developer.mozilla.org/en/docs/Web/API/SVGTextElement) to Kotlin
 */
public external abstract class SVGTextElement : SVGElement, SVGURIReference {
    open val x: SVGAnimatedLength
    open val y: SVGAnimatedLength
    open val dx: SVGAnimatedLength
    open val dy: SVGAnimatedLength

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [SVGTextPathElement](https://developer.mozilla.org/en/docs/Web/API/SVGTextPathElement) to Kotlin
 */
public external abstract class SVGTextPathElement : SVGElement, SVGURIReference {
    open val x: SVGAnimatedLength
    open val y: SVGAnimatedLength
    open val dx: SVGAnimatedLength
    open val dy: SVGAnimatedLength
    open val align: SVGTextAnchorType
    open val viewBox: SVGAnimatedLength
    open val pathLength: SVGAnimatedNumber
    open val rotation: SVGAnimatedNumber

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [SVGStyleElement](https://developer.mozilla.org/en/docs/Web/API/SVGStyleElement) to Kotlin
 */
public external abstract class SVGStyleElement : SVGElement, SVGURIReference {
    open val x: SVGAnimatedLength
    open val y: SVGAnimatedLength
    open val dx: SVGAnimatedLength
    open val dy: SVGAnimatedLength

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [SVGSVGElement](https://developer.mozilla.org/en/docs/Web/API/SVGSVGElement) to Kotlin
 */
public external abstract class SVGSVGElement : SVGElement, SVGURIReference {
    open val x: SVGAnimatedLength
    open val y: SVGAnimatedLength
    open val width: SVGAnimatedLength
    open val height: SVGAnimatedLength
    open val viewBox: SVGAnimatedLength
    open val preserveAspectRatio: SVGPreserveAspectRatio

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
Short
}|n|\n\n/* Exposes the JavaScript
[FileReaderSync](https://developer.mozilla.org/en/docs/Web/API/FileReaderSync) to Kotlin
*/

```kotlin
fun readAsArrayBuffer(blob: Blob): ArrayBuffer
fun readAsText(blob: Blob, label: String = definedExternally): String
fun readAsDataURL(blob: Blob): String
```

Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

// NOTE: THIS FILE IS AUTO-GENERATED, DO NOT EDIT! See

 Advance the execution of the code

```kotlin
public external
```

```kotlin
open class FileReaderSync {
    fun readAsArrayBuffer(blob: Blob): ArrayBuffer
    fun readAsBinaryString(blob: Blob): String
    fun readAsText(blob: Blob, label: String = definedExternally): String
    fun readAsDataURL(blob: Blob): String
}
```

```kotlin
/*
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */
```

```kotlin
// NOTE: THIS FILE IS AUTO-GENERATED, DO NOT EDIT!
// See

package org.w3c.notifications

```

```kotlin
import kotlin.js.*
import org.khronos.webgl.*
import org.w3c.css.masking.*
import org.w3c.dom.*
import org.w3c.dom.clipboard.*
import org.w3c.dom.css.*
import org.w3c.dom.encryptedmedia.*
import org.w3c.dom.events.*
import org.w3c.dom.mediacapture.*
import org.w3c.dom.mediasource.*
import org.w3c.dom.parsing.*
import org.w3c.dom.pointerevents.*
import org.w3c.dom.svg.*
import org.w3c.dom.url.*
import org.w3c.fetch.*
import org.w3c.files.*
import org.w3c.performance.*
import org.w3c.workers.*
import org.w3c.xhr.*
```

```kotlin
/**
 * Exposes the JavaScript
[Notification](https://developer.mozilla.org/en/docs/Web/API/Notification) to Kotlin
 */
public external open
```

```kotlin
class Notification(title: String, options: NotificationOptions = definedExternally) : EventTarget {
    var onclick: ((MouseEvent) -> dynamic)?
    var onerror: ((Event) -> dynamic)?
    open val title: String
    open val dir:
    NotificationDirection
    open val lang: String
    open val body: String
    open val tag: String
    open val image: String
    open val icon: String
    open val badge: String
    open val sound: String
    open val vibrate: dynamic
    open val timestamp: Number
    open val renotify: Boolean
    open val silent: Boolean
    open val noscreen: Boolean
    open val requireInteraction: Boolean
    open val open: String?
    open val close(): Boolean
    open val config: NotificationConfig?
    open val pending: NotificationPending?
    open val id: String
    open val gid: String
    open val actions: Array<NotificationAction>
    open val origin: String?
    open val revocationKey: String?
    open val priority: String?
    open val delayedUntil: Number?
    open val canceledDueToPermission: Boolean?
    open val canceledDueToUserAcceptance: Boolean?
    open val canceledDueToUserTiming: Boolean?
    open val canceledDueToUserDelay: Boolean?
    open val canceledDueToUserOverlap: Boolean?
    open val canceledDueToUserReason: String?
    open val canceledDueToUserMessage: String?
    fun close()

    companion object {
        val permission: NotificationPermission
        val maxActions: Int
        fun requestPermission(deprecatedCallback: (NotificationPermission) -> Unit = definedExternally): Promise<NotificationPermission>
    }
}
```

```kotlin
public external interface NotificationOptions {
    var dir: NotificationDirection? /* = NotificationDirection.AUTO */
    var lang: String?
    var body: String?
    var tag: String?
    var image: String?
    var icon: String?
    var badge: String?
    var sound: String?
    var vibrate: dynamic
    var timestamp: Number?
    var renotify: Boolean?
    var silent: Boolean?
    var noscreen: Boolean?
    var requireInteraction: Boolean?
    var sticky: Boolean?
    var data: Any?
    var actions: Array<NotificationAction>?
}
```

```kotlin
@kotlin.internal.InlineOnly
public inline fun NotificationOptions(dir: NotificationDirection? = NotificationDirection.AUTO, lang: String? = "", body:
```

```kotlin
```

```kotlin
fun asObject(): NotificationOptions?
```

```kotlin
fun asArray(): Any
```

```kotlin
fun asMap(): Map<String, Any>
```

```kotlin
fun asMap(key: String, value: Any): Map<String, Any>
```

```kotlin
fun asMap(key: String, value: Any): Map<String, Any>
```

```kotlin
fun asArray(): Any
```

```kotlin
fun asMap(): Map<String, Any>
```

```kotlin
fun asMap(key: String, value: Any): Map<String, Any>
```

```kotlin
fun asArray(): Any
```

```kotlin
fun asMap(): Map<String, Any>
```

```kotlin
fun asMap(key: String, value: Any): Map<String, Any>
```

```kotlin
fun asArray(): Any
```

```kotlin
fun asMap(): Map<String, Any>
```

```kotlin
fun asMap(key: String, value: Any): Map<String, Any>
```

```kotlin
fun asArray(): Any
```

```kotlin
fun asMap(): Map<String, Any>
```

```kotlin
fun asMap(key: String, value: Any): Map<String, Any>
```

```kotlin
fun asArray(): Any
```

```kotlin
fun asMap(): Map<String, Any>
```

```kotlin
fun asMap(key: String, value: Any): Map<String, Any>
```

```kotlin
fun asArray(): Any
```
```kotlin
    return NotificationAction {
        "action": action
    }
}

open fun NotificationPermission(): NotificationPermission {
    return NotificationPermission.DEFAULT
}

open fun NotificationDirection(): NotificationDirection {
    return NotificationDirection.AUTO
}
```

---

```
/*
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

// NOTE: THIS FILE IS AUTO-GENERATED, DO NOT EDIT!
// See github.com/kotlin/dukat for details

package org.w3c.workers

import kotlin.js.*
import org.khronos.webgl.*
import org.w3c.css.masking.*
import org.w3c.dom.*
import org.w3c.dom.clipboard.*
import org.w3c.dom.css.*
import org.w3c.dom.encryptedmedia.*
import org.w3c.dom.events.*
import org.w3c.dom.mediacapture.*
import org.w3c.dom.mediasource.*
import org.w3c.dom.parsing.*
import org.w3c.dom.pointerevents.*
import org.w3c.dom.svg.*
import org.w3c.dom.url.*
import org.w3c.fetch.*
import org.w3c.files.*
import org.w3c.notifications.*
import org.w3c.performance.*
import org.w3c.xhr.*
```

---

```
/**
 * Exposes the JavaScript [NotificationEvent](https://developer.mozilla.org/en/docs/Web/API/NotificationEvent) to Kotlin
 */
public external open class NotificationEvent(type: String, eventInitDict: NotificationEventInit) : ExtendableEvent {

    open val notification: Notification
    open val action: String

    companion object {

        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }

    public external interface NotificationEventInit : ExtendableEventInit {

        var notification: Notification?
        var action: String
    }

    public inline val NotificationEventInit.notification: Notification
    public inline val NotificationEventInit.action: String
```

---

```
/* please, don’t implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface NotificationPermission {

    companion object
}

public inline val NotificationPermission.Companion.DEFAULT: NotificationPermission get() = "default".asDynamic().unsafeCast<NotificationPermission>()

public inline val NotificationPermission.Companion.DENIED: NotificationPermission get() = "denied".asDynamic().unsafeCast<NotificationPermission>()

public inline val NotificationPermission.Companion.GRANTED: NotificationPermission get() = "granted".asDynamic().unsafeCast<NotificationPermission>()

/* please, don’t implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface NotificationDirection {

    companion object
}

public inline val NotificationDirection.Companion.AUTO: NotificationDirection get() = "auto".asDynamic().unsafeCast<NotificationDirection>()

public inline val NotificationDirection.Companion.LTR: NotificationDirection get() = "ltr".asDynamic().unsafeCast<NotificationDirection>()

public inline val NotificationDirection.Companion.RTL: NotificationDirection get() = "rtl".asDynamic().unsafeCast<NotificationDirection>()
```

---

```
/*
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

// NOTE: THIS FILE IS AUTO-GENERATED, DO NOT EDIT!
// See github.com/kotlin/dukat for details

package org.w3c.workers

import kotlin.js.*
import org.khronos.webgl.*
import org.w3c.css.masking.*
import org.w3c.dom.*
import org.w3c.dom.clipboard.*
import org.w3c.dom.css.*
import org.w3c.dom.encryptedmedia.*
import org.w3c.dom.events.*
import org.w3c.dom.mediacapture.*
import org.w3c.dom.mediasource.*
import org.w3c.dom.parsing.*
import org.w3c.dom.pointerevents.*
import org.w3c.dom.svg.*
import org.w3c.dom.url.*
import org.w3c.fetch.*
import org.w3c.files.*
import org.w3c.notifications.*
import org.w3c.performance.*
import org.w3c.xhr.*
```

---
abstract class ServiceWorker : EventTarget, AbstractWorker, UnionMessagePortOrServiceWorker, UnionClientOrMessagePortOrServiceWorker {
    open val scriptURL: String
    open var onstatechange: ((Event) -> dynamic)?
    fun postMessage(message: Any?, transfer: Array<dynamic> = definedExternally) {
        // implementation
    }
}

/**
 * Exposes the JavaScript
[ServiceWorkerRegistration](https://developer.mozilla.org/en/docs/Web/API/ServiceWorkerRegistration) to Kotlin
 */
public external abstract class ServiceWorkerRegistration : EventTarget {
    open val installing: ServiceWorker?
    open val waiting: ServiceWorker?
    open val active: ServiceWorker?
    open val scope: String
    open var onupdatefound: ((Event) -> dynamic)?
    open val APISpace: dynamic
    fun update(): Promise<Unit>
    fun unregister(): Promise<Boolean>
    fun showNotification(title: String, options: NotificationOptions = definedExternally): Promise<void>
    fun getNotifications(filter: GetNotificationOptions = definedExternally): Promise<Array<Notification>>
    fun methodName(): Promise<dynamic>
}

/**
 * Exposes the JavaScript
[ServiceWorkerContainer](https://developer.mozilla.org/en/docs/Web/API/ServiceWorkerContainer) to Kotlin
 */
public external abstract class ServiceWorkerContainer : EventTarget {
    open val controller: ServiceWorker?
    open val ready: Promise<ServiceWorkerRegistration>
    open var oncontrollerchange: ((Event) -> dynamic)?
    open var onmessage: ((MessageEvent) -> dynamic)?
    fun register(scriptURL: String, options: RegistrationOptions = definedExternally): Promise<ServiceWorkerRegistration>
    fun getRegistration(clientURL: String = definedExternally): Promise<ServiceWorkerRegistration>
    fun startMessages(): Promise<Unit>
    fun getRegistrations(): Promise<Array<ServiceWorkerRegistration>>
}

public external interface RegistrationOptions {
    var scope: String?
        get() = definedExternally
        set(value) = definedExternally
    var type: WorkerType? /* = WorkerType.CLASSIC */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun RegistrationOptions(scope: String? = undefined, type: WorkerType? = WorkerType.CLASSIC): RegistrationOptions {
    val o = js("({})")
    o["scope"] = scope
    o["type"] = type
    return o
}

/**
 * Exposes the JavaScript
[ServiceWorkerMessageEvent](https://developer.mozilla.org/en/docs/Web/API/ServiceWorkerMessageEvent) to Kotlin
 */
public external open class ServiceWorkerMessageEvent(type: String, eventInitDict: ServiceWorkerMessageEventInit = definedExternally) : Event {
    open val data: Any?
    open val origin: String
    open val lastEventId: String
    open var source: UnionMessagePortOrServiceWorker?
    open var ports: Array<MessagePort>?

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public external interface ServiceWorkerMessageEventInit : EventInit {
    var data: Any?
        get() = definedExternally
        set(value) = definedExternally
    var origin: String?
        get() = definedExternally
        set(value) = definedExternally
    var lastEventId: String?
        get() = definedExternally
        set(value) = definedExternally
    var source: UnionMessagePortOrServiceWorker?
        get() = definedExternally
        set(value) = definedExternally
    var ports: Array<MessagePort>?
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["data"] = data
    o["origin"] = origin
    o["lastEventId"] = lastEventId
    o["source"] = source
    o["ports"] = ports
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}

/**
 * Exposes the JavaScript
[ServiceWorkerGlobalScope](https://developer.mozilla.org/en/docs/Web/API/ServiceWorkerGlobalScope) to Kotlin
 */
public external abstract class ServiceWorkerGlobalScope : WorkerGlobalScope {
    open val clients: Clients
    open val registration: ServiceWorkerRegistration
    open var oninstall: ((Event) -> dynamic)?
    open var onactivate: ((Event) -> dynamic)?
    open var onfetch: ((FetchEvent) -> dynamic)?
    open var onerror: ((Event) -> dynamic)?
    open var onmessage: ((MessageEvent) -> dynamic)?
    open var onnotificationclick: ((NotificationEvent) -> dynamic)?
    open var onnotificationclose: ((NotificationEvent) ->}
fun onfunctionalevent: ((Event) -> dynamic)?
fun skipWaiting():
Promise<Unit>?

open var onfunctionalevent: ((Event) -> dynamic)?
fun onfunctionalevent: ((Event) -> dynamic)?
fun skipWaiting():
Promise<Unit>

/**
 * Exposes the JavaScript [Client](https://developer.mozilla.org/en/docs/Web/API/Client) to Kotlin
 */
public external abstract class Client : UnionClientOrMessagePortOrServiceWorker {
    open val url: String
    open val frameType: FrameType
    open val id: String
    fun postMessage(message: Any?, transfer: Array<dynamic> = definedExternally)
}

/**
 * Exposes the JavaScript [WindowClient](https://developer.mozilla.org/en/docs/Web/API/WindowClient) to Kotlin
 */
public external abstract class WindowClient : Client {
    open val visibilityState: dynamic
    open val focused: Boolean
    fun focus(): Promise<WindowClient>
    fun navigate(url: String): Promise<WindowClient>
}

/**
 * Exposes the JavaScript [Clients](https://developer.mozilla.org/en/docs/Web/API/Clients) to Kotlin
 */
public external abstract class Clients {
    fun get(id: String): Promise<Any?>
    fun matchAll(options: ClientQueryOptions = definedExternally): Promise<Array<Client>>
    fun openWindow(url: String): Promise<WindowClient?>
    fun claim(): Promise<Unit>
}

public external interface ClientQueryOptions {
    var includeUncontrolled: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var type: ClientType? /* = ClientType.WINDOW */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun ClientQueryOptions(includeUncontrolled: Boolean? = false, type: ClientType? = ClientType.WINDOW): ClientQueryOptions {
    val o = js("({})")
    o["includeUncontrolled"] = includeUncontrolled
    o["type"] = type
    return o
}

/**
 * Exposes the JavaScript [ExtendableEvent](https://developer.mozilla.org/en/docs/Web/API/ExtendableEvent) to Kotlin
 */
public external open class ExtendableEvent(type: String, eventInitDict: ExtendableEventInit = definedExternally) : Event {
    fun waitUntil(f: Promise<Any?>)

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public external interface ExtendableEventInit : EventInit

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}

/**
 * Exposes the JavaScript [InstallEvent](https://developer.mozilla.org/en/docs/Web/API/InstallEvent) to Kotlin
 */
public external open class InstallEvent(type: String, eventInitDict: ExtendableEventInit = definedExternally) : ExtendableEvent {
    fun registerForeignFetch(options: ForeignFetchOptions)

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public external interface ForeignFetchOptions {
    var scopes: Array<String>?
        get() = definedExternally
        set(value) = definedExternally
    var origins: Array<String>?
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun ForeignFetchOptions(scopes: Array<String>?, origins: Array<String>?): ForeignFetchOptions {
    val o = js("({})")
    o["scopes"] = scopes
    o["origins"] = origins
    return o
}

/**
 * Exposes the JavaScript [FetchEvent](https://developer.mozilla.org/en/docs/Web/API/FetchEvent) to Kotlin
 */
public external open class FetchEvent(type: String, eventInitDict: FetchEventInit) : ExtendableEvent {
    open val request: Request
    open val clientId: String?
    open val isReload: Boolean
    fun respondWith(r: Promise<Response>)

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public external interface FetchEventInit : ExtendableEventInit {
    var request: Request?
        get() = definedExternally
        set(value) = definedExternally
    var clientId: String? /* = null */
        get() = definedExternally
        set(value) = definedExternally
    var isReload: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["request"] = request
    o["clientId"] = clientId
    o["isReload"] = isReload
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}

/**
 * Exposes the JavaScript [ForeignFetchEvent](https://developer.mozilla.org/en/docs/Web/API/ForeignFetchEvent) to Kotlin
 */
public external open class ForeignFetchEvent(type: String, eventInitDict: ForeignFetchEventInit) : ExtendableEvent {
    fun open(val request: Request?)
}
import org.w3c.dom.parsing.*
import org.w3c.dom.pointerevents.*
import org.w3c.dom.svg.*
import org.w3c.dom.url.*
import org.w3c.fetch.*
import org.w3c.files.*
import org.w3c.notifications.*
import org.w3c.performance.*
import org.w3c.workers.*

/**
 * Exposes the JavaScript
 [XMLHttpRequestEventTarget](https://developer.mozilla.org/en/docs/Web/API/XMLHttpRequestEventTarget) to Kotlin
 */
public external abstract class XMLHttpRequestEventTarget : EventTarget {
    open var onloadstart:
        ((ProgressEvent) -> dynamic)?
    open var onprogress: ((ProgressEvent) -> dynamic)?
    open var onabort:
        ((Event) -> dynamic)?
    open var onerror: ((Event) -> dynamic)?
    open var onload: ((Event) -> dynamic)?
    open var ontimeout: ((Event) -> dynamic)?
    open var onloadend: ((Event) -> dynamic)?
}

public external abstract class XMLHttpRequestUpload : XMLHttpRequestEventTarget

/**
 * Exposes the JavaScript
 [XMLHttpRequest](https://developer.mozilla.org/en/docs/Web/API/XMLHttpRequest) to Kotlin
 */
public external open class XMLHttpRequest : XMLHttpRequestEventTarget {
    var onreadystatechange: ((Event) -> dynamic)?
    open val readyState: Short
    var timeout: Int
    var withCredentials: Boolean
    open val upload: XMLHttpRequestUpload
    open val responseURL: String
    open val status: Short
    open val statusText: String
    var responseType: XMLHttpRequestResponseType
    open val response: Any?
    open val responseText: String
    open val responseXML: Document?
    fun open(method: String, url: String)
    fun open(method: String, url: String, async: Boolean, username: String? = definedExternally, password: String? =
        definedExternally)
    fun setRequestHeader(name: String, value: String)
    fun send(body: dynamic = definedExternally)
    fun abort()
    fun getResponseHeader(name: String): String?
    fun getAllResponseHeaders(): String
    fun overrideMimeType(mime: String)

    companion object {
        val UNSENT: Short
        val OPENED: Short
        val HEADERS_RECEIVED: Short
        val LOADING: Short
        val DONE: Short
    }
}

/**
 * Exposes the JavaScript
 [FormData](https://developer.mozilla.org/en/docs/Web/API/FormData) to Kotlin
 */
public external open class FormData(form: HTMLFormElement = definedExternally) {
    fun append(name: String, value: String)
    fun append(name: String, value: Blob, filename: String = definedExternally)
    fun delete(name: String)
    fun get(name: String): dynamic
    fun getAll(name: String): Array<dynamic>
    fun has(name: String): Boolean
    fun set(name: String, value: String)
    fun set(name: String, value: Blob, filename: String =
        definedExternally)
}

/**
 * Exposes the JavaScript
 [ProgressEvent](https://developer.mozilla.org/en/docs/Web/API/ProgressEvent) to Kotlin
 */
public external open class ProgressEvent(type: String, eventInitDict: ProgressEventInit = definedExternally) :
    Event {
    open val lengthComputable: Boolean
    open val loaded: Number
    open val total: Number

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public external interface ProgressEventInit : EventInit {
    var lengthComputable: Boolean? /* = false */
    var loaded: Number? /* = 0 */
    var total: Number? /* = 0 */
    var bubbles: Boolean? /* = false */
    var cancelable: Boolean? /* = false */
}

@kotlin.internal.InlineOnly
public inline fun ProgressEventInit(lengthComputable:
    Boolean? = false, loaded: Number? = 0, total: Number? = 0, bubbles: Boolean? = false, cancelable: Boolean? =
    false): ProgressEventInit {
    var lengthComputable: Boolean? = false,
    loaded: Number? = 0, total: Number? = 0, bubbles: Boolean? = false,
    cancelable: Boolean? = false):
    ProgressEventInit {
        "lengthComputable": lengthComputable
    }
}

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface XMLHttpRequestResponseType {
    companion object
}

public inline val XMLHttpRequestResponseType.Companion.EMPTY: XMLHttpRequestResponseType get() =
    "".asDynamic().unsafeCast<XMLHttpRequestResponseType>()

public inline val XMLHttpRequestResponseType.Companion.ARRAYBUFFER: XMLHttpRequestResponseType get() =
    "arraybuffer".asDynamic().unsafeCast<XMLHttpRequestResponseType>()

public inline val XMLHttpRequestResponseType.Companion.BLOB: XMLHttpRequestResponseType get() =
    "blob".asDynamic().unsafeCast<XMLHttpRequestResponseType>()
XMLHttpRequestResponseType.Companion.DOCUMENT: XMLHttpRequestResponseType get() = "document".asDynamic().unsafeCast<XMLHttpRequestResponseType>()

XMLHttpRequestResponseType.Companion.JSON: XMLHttpRequestResponseType get() = "json".asDynamic().unsafeCast<XMLHttpRequestResponseType>()

XMLHttpRequestResponseType.Companion.TEXT: XMLHttpRequestResponseType get() = "text".asDynamic().unsafeCast<XMLHttpRequestResponseType>()

/*
 * Copyright 2010-2020 JetBrains s.r.o.
 * and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license
 * that can be found in the license/LICENSE.txt file.
 */

package kotlin

/**
 * Creates a Char with the specified [code], or throws an exception if the [code] is out of
 * `Char.MIN_VALUE.`
 * If the program that calls this function is written in a way that only valid [code] is passed as the argument,
 * using the overload that takes a [UShort] argument is preferable (`Char(intValue.toUShort())`).
 * That overload doesn't check validity of the argument, and may improve program performance when the function is called
 * routinely inside a loop.
 * @sample samples.text.Chars.charFromCode
 */
@OptIn(ExperimentalUnsignedTypes::class)
@ExperimentalStdlibApi
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun Char(code: Int): Char {
    if (code < Char.MIN_VALUE || code > Char.MAX_VALUE) {
        throw IllegalArgumentException("Invalid Char code: $code")
    }
    return Char(code.toUShort())
}

/**
 * Creates a Char with the specified [code].
 * @sample samples.text.Chars.charFromCode
 */
@OptIn(ExperimentalUnsignedTypes::class)
@ExperimentalStdlibApi
@SinceKotlin("1.4")
@Suppress("NO_ACTUAL_FOR_EXPECT")
public expect fun Char(code: UShort): Char

/**
 * Returns the code of this Char.
 * Code of a Char is the value it was constructed with, and the UTF-16 code unit corresponding to this Char.
 * @sample samples.text.Chars.charFromCode
 */
@ExperimentalStdlibApi
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline val Char.code: Int get() = this.toInt()

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin

import kotlin.annotation.AnnotationRetention.BINARY
import kotlin.annotation.AnnotationRetention.SOURCE
import kotlin.internal.RequireKotlin
import kotlin.reflect.KClass

/**
 * Signals that the annotated annotation class is a marker of an experimental API.
 * Any declaration annotated with that marker is considered an experimental declaration
 * and its call sites should accept the experimental aspect of it either by using
 * [UseExperimental], or by being annotated with that marker themselves, effectively causing further
 * propagation of that experimental aspect.
 * This class is deprecated in favor of a more general approach provided by [RequiresOptIn]/[OptIn].
 */
@Target(ANNOTATION_CLASS)
@Retention(BINARY)
@SinceKotlin("1.2")
@RequiresOptIn("Please use RequiresOptIn instead.")
@Deprecated("Please use RequiresOptIn instead.")

/**
 * Allows to use experimental API denoted by the given markers in the annotated file, declaration, or expression.
 * This class is deprecated in favor of a more general approach provided by [RequiresOptIn]/[OptIn].
 */
@Target(CLASS, PROPERTY, LOCAL_VARIABLE, VALUE_PARAMETER, CONSTRUCTOR, FUNCTION, PROPERTY_GETTER, PROPERTY_SETTER, EXPRESSION, FILE,
Apache 2.0 license that can be found in the license/LICENSE.txt file.

```kotlin
package kotlin

import kotlin.annotation.AnnotationTarget
import kotlin.internal.RequireKotlin
import kotlin.internal.RequireKotlinVersionKind

/**
 * The experimental multiplatform support API marker.
 * Any usage of a declaration annotated with `@ExperimentalMultiplatform` must be accepted either by annotating that usage with the [OptIn] annotation, e.g. `@OptIn(ExperimentalMultiplatform::class)`.
 * or by using the compiler argument `-Xopt-in=kotlin.ExperimentalMultiplatform`.
 */
@Suppress("DEPRECATION")
@Experimental
@RequiresOptIn
@MustBeDocumented
@Target(
    CLASS, ANNOTATION_CLASS, PROPERTY, FIELD, LOCAL_VARIABLE, VALUE_PARAMETER, CONSTRUCTOR, FUNCTION, PROPERTY_GETTER, PROPERTY_SETTER, TYPEALIAS)
@Retention(AnnotationRetention.BINARY)
@RequireKotlin("1.2.50", versionKind = RequireKotlinVersionKind.COMPLIER_VERSION)
public annotation class ExperimentalMultiplatform

/**
 * Marks an expected annotation class that it isn’t required to have actual counterparts in all platforms.
 * This annotation is only applicable to ‘expect’ annotation classes in multi-platform projects and marks that class as optional.
 * Optional expected class is allowed to have no corresponding actual class on the platform. Optional annotations can only be used to annotate something, not as types in signatures. If an optional annotation has no corresponding actual class on a platform, the annotation entries where it’s used are simply erased when compiling code on that platform.
 * Note: this annotation is experimental, see [ExperimentalMultiplatform] on how to opt-in for it.
 */
@Target(ANNOTATION_CLASS)
@Retention(AnnotationRetention.BINARY)
@ExperimentalMultiplatform
@RequireKotlinVersionKind.COMPLIER_VERSION
public annotation class OptionalExpectation

/**
 * Signals that the annotated annotation class is a marker of an API that requires an explicit opt-in.
 * Call sites of any declaration annotated with that marker should opt in to the API either by using [OptIn], or by being annotated with that marker themselves, effectively causing further propagation of the opt-in requirement.
 * This class requires opt-in itself and can only be used with the compiler argument `-Xopt-in=kotlin.RequiresOptIn`.
 */
@Target(CLASS, PROPERTY, LOCAL_VARIABLE, VALUE_PARAMETER, CONSTRUCTOR, FUNCTION, PROPERTY_GETTER, PROPERTY_SETTER, TYPEALIAS)
@Retention(SOURCE)
@SinceKotlin("1.3")
@RequireKotlin("1.3.70", versionKind = RequireKotlinVersionKind.COMPLIER_VERSION)
public annotation class RequiresOptIn {
    /**
     * Severity of the diagnostic that should be reported on usages which did not explicitly opted into the API either by using [OptIn] or by being annotated with that marker
     */
    public enum class Level {
        /** Specifies that a warning should be reported on incorrect usages of this API. */
        WARNING,

        /** Specifies that an error should be reported on incorrect usages of this API. */
        ERROR,
    }
}

/**
 * Allows to use the API denoted by the given markers in the annotated file, declaration, or expression.
 * If a declaration is annotated with [OptIn], its usages are not required to opt in to that API.
 * This class requires opt-in itself and can only be used with the compiler argument `-Xopt-in=kotlin.RequiresOptIn`.
 */
@Target(CLASS, PROPERTY, LOCAL_VARIABLE, VALUE_PARAMETER, CONSTRUCTOR, FUNCTION, PROPERTY_GETTER, PROPERTY_SETTER, EXPRESSION, FILE, TYPEALIAS)
@Retention(SOURCE)
@SinceKotlin("1.3")
@RequireKotlin("1.3.70", versionKind =
```
RequireKotlinVersionKind.COMPILER_VERSION

public annotation class OptIn
(vararg val markerClass: KClass<out Annotation>)

/*
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language
 * contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in
 * the license/LICENSE.txt file. *
 */

package kotlin.collections

import kotlin.js.JsName

/**
 * Provides a skeletal implementation of the read-only [Collection] interface. *
 * Elements are stored in a backing array.
 */
@SinceKotlin("1.1")
public abstract class
AbstractCollection<out E> protected constructor() : Collection<E>()

abstract override fun iterator(): Iterator<E>

override fun contains(element: @UnsafeVariance E): Boolean = any { it == element }

override fun containsAll(elements: Collection<@UnsafeVariance E>): Boolean =
elements.all { contains(it) } *

override fun toString(): String = joinToString(", ", "[", "]") {
    if (it === this) "(this Collection)" else it.toString()
}

/**
 * Returns new array of type `Array<Any?>` with the elements of this collection.
 * * @JsName("toArray")
 */
@JsName("toArray")
protected open fun toArray(): Array<Any?> = copyToArrayImpl(this)

protected open fun <T> toArray(array: Array<T>): Array<T> = copyToArrayImpl(this, array)

/**
 * A base class to simplify implementing iterators so that implementations only have to implement [computeNext]
 * to implement the iterator, calling [done] when the iteration is complete. *
 */
public abstract class AbstractIterator<T> : Iterator<T> {
    private var state = State.NotReady
    private var nextValue: T? = null

    override fun hasNext(): Boolean {
        require(state != State.Failed)
        return when (state) {
            State.Done -> false
            State.Ready -> true
            else -> tryToComputeNext()
        }
    }

    override fun next(): T {
        if (!hasNext()) throw NoSuchElementException()
        state = State.Ready
        @Suppress("UNCHECKED_CAST")
        return nextValue as T
    }

    private fun tryToComputeNext(): Boolean {
        state = State.Failed
        computeNext()
        return state == State.Ready
    }

    /**
     * Computes the next item in the iteration, called from the [computeNext] function
     */
    abstract protected fun computeNext(): Unit

    /**
     * Sets the next value in the iteration, called from the [computeNext] function
     */
    protected fun setNext(value: T): Unit {
        nextValue = value
        state = State.Ready
    }

    /**
     * Sets the state to done so that the iteration terminates. *
     */
    protected fun done() {
        state = State.Done
    }

    /**********
    * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
    * Use of this source code is governed by the Apache 2.0 license that can be found in
    * the license/LICENSE.txt file. *
    */

    private enum class State {
        Ready, NotReady, Done, Failed
    }

    private class SubList<out E>(private val list: AbstractList<E>, private val fromIndex: Int, toIndex: Int) : AbstractList<E>(), RandomAccess {
        private var _size: Int = 0

        init {
            checkRangeIndexes(fromIndex, toIndex, list.size)
        }

        override fun get(index: Int): E {
            return list.get(index)
        }

        override fun iterator(): Iterator<E> = list.iterator()

        override fun subList(fromIndex: Int, toIndex: Int): List<E> = SubList(fromIndex, toIndex, list.size)

    }
Open Source Used In security_automation identity-guest-cs  1753

```java
public class ArrayDeque<E> : AbstractMutableList<E> {
    private var head: Int = 0
    private var elementData: Array<Any?>

    override var size: Int = 0
        private set

    /**
     * Constructs an empty deque with specified [initialCapacity], or
     * throws [IllegalArgumentOutOfRangeException] if [initialCapacity] is negative.
     */
    public constructor(initialCapacity: Int) {
        elementData = when {
            initialCapacity == 0 -> emptyElementData
            initialCapacity > 0 -
            > arrayOfNulls(initialCapacity)
            else -> throw IllegalArgumentException("Illegal Capacity:
            \n```
```kotlin
/**
 * Returns `true` if this nullable array is either null or empty.
 * @sample samples.collections.Arrays.Usage.arrayIsNullOrEmpty
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun Array<*>?.isNullOrEmpty(): Boolean {
    contract {
        returns(false) implies (this@isNullOrEmpty != null)
    }
    return this == null || this.isEmpty()
}

/**
 * Returns `true` if this nullable array is either null or empty.
 * @sample samples.collections.Arrays.Usage.arrayIsNullOrEmpty
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
@Suppress("UPPER_BOUND_CANNOT_BE_ARRAY")
public inline fun <C, R> C.ifEmpty(defaultValue: () -> R): R where C : Array<*>, C : R =
    if (isNullOrEmpty()) defaultValue() else
    this

@OptIn(ExperimentalUnsignedTypes::class)
@SinceKotlin("1.3")
@PublishedApi
@kotlin.jvm.JvmName("contentDeepEquals")
@kotlin.js.JsName("contentDeepEqualsImpl")
internal fun <T> Array<out T>?.contentDeepEqualsImpl(other: Array<out T>?): Boolean {
    if (this === other) return true
    if (this == null || other == null) return false
    if (this.size != other.size) return false
    for (i in indices) {
        val v1 = this[i]
        val v2 = other[i]
        if (v1 == null || v2 == null) return false
        when {
            v1 is Array<*>, v2 is Array<*> -> if (!v1.contentEquals(v2)) return false
            v1 is ByteArray, v2 is ByteArray -> if (!v1.contentEquals(v2)) return false
            v1 is ShortArray, v2 is ShortArray -> if (!v1.contentEquals(v2)) return false
            v1 is IntArray, v2 is IntArray -> if (!v1.contentEquals(v2)) return false
            v1 is LongArray, v2 is LongArray -> if (!v1.contentEquals(v2)) return false
            v1 is FloatArray, v2 is FloatArray -> if (!v1.contentEquals(v2)) return false
            v1 is DoubleArray, v2 is DoubleArray -> if (!v1.contentEquals(v2)) return false
            v1 is CharArray, v2 is CharArray -> if (!v1.contentEquals(v2)) return false
            v1 is BooleanArray, v2 is BooleanArray -> if (!v1.contentEquals(v2)) return false
            v1 is UByteArray, v2 is UByteArray -> if (!v1.contentEquals(v2)) return false
            v1 is UShortArray, v2 is UShortArray -> if (!v1.contentEquals(v2)) return false
            v1 is UIntArray, v2 is UIntArray -> if (!v1.contentEquals(v2)) return false
            v1 is ULongArray, v2 is ULongArray -> if (!v1.contentEquals(v2)) return false
            else -> if (v1 != v2) return false
        }
    }
    return true
}

@SinceKotlin("1.3")
@PublishedApi
@kotlin.jvm.JvmName("contentDeepToString")
@kotlin.js.JsName("contentDeepToStringImpl")
internal fun <T> Array<out T>?.contentDeepToStringImpl(): String {
    if (this == null) return "null"
    val length = size.coerceAtMost((Int.MAX_VALUE - 2) / 5) * 5 + 2 // in order not to overflow Int.MAX_VALUE
    return buildString(length) {
        contentDeepToStringInternal(this, mutableListOf())
    }
}

@OptIn(ExperimentalUnsignedTypes::class)
private fun <T> Array<out T>.contentDeepToStringInternal(result: StringBuilder, processed: MutableList<Array<*>>) {
    if (this in processed) {
        result.append("[...]")
        return
    }
    processed.add(this)
    result.append('[')
    for (i in indices) {
        if (i != 0) {
            result.append(', ')
        }
        val element = this[i]
        when (element) {
            null -> result.append('null')
            is Array<*>, is DoubleArray, is CharArray, is BooleanArray ->
                result.append(element.contentToString())
            is UByteArray, is UShortArray, is UIntArray, is ULongArray ->
                result.append(element.contentToString())
            else ->
                result.append(element.toString())
        }
    }
    result.append(']')
    processed.removeAt(processed.lastIndex)
}
```

```
/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.collections

```
IndexedValue<out T>(public val index: Int, public val value: T)

* @param thisRef the object for which the value is requested (not used).
* @param property the metadata for the property, used to get the name of property and lookup the value corresponding to this name in the map.
* @return the property value.
* @throws NoSuchElementException when the map doesn't contain value for the property name and doesn't provide an implicit default (see [withDefault]).

@kotlin.internal.InlineOnly
public inline operator fun <V, V1 : V> Map<in String, @Exact V>.getValue(thisRef: Any?, property: KProperty<*>): V1 =
    @Suppress("UNCHECKED_CAST") (getOrImplicitDefault(property.name) as V1)

/**
 * Stores the value of the property for the given object in this mutable map.
* @throws NoSuchElementException when the map doesn't contain value for the property name and no implicit default was provided for that map.
* @param property the metadata for the property, used to get the name of property and store the value associated with that name in the map.
* @param value the value to set.
*/
@kotlin.internal.InlineOnly
public inline operator fun <V> MutableMap<in String, in V>.setValue(thisRef: Any?, property: KProperty<*>, value: V)
    {
        this.put(property.name, value)
    }

@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("MapsKt")

package kotlin.collections

/**
 * Returns a wrapper of this read-only map, having the implicit default value provided with the specified function [defaultValue].
 * This implicit default value is used when the original map doesn't contain a value for the key specified and no implicit default was provided for that map.
 * @param defaultValue the function that is used to provide the default value for each key.
 */
@kotlin.jvm.JvmName("withDefaultMutable")
public fun <K, V> MutableMap<K, V>.withDefault(defaultValue: (key: K) -> V): MutableMap<K, V> =
    when (this)
    {
        is MapWithDefault ->
            this.map.withDefault(defaultValue)
        else -> MapWithDefaultImpl(this, defaultValue)
    }

/**
 * Stores the value of the property for the given object in this mutable map.
* @param thisRef the object for which the value is requested (not used).
* @param property the metadata for the property, used to get the name of property and store the value associated with that name in the map.
* @param value the value to set.
*/
@kotlin.internal.InlineOnly
public inline operator fun <V> MutableMap<in String, in V>.setValue(thisRef: Any?, property: KProperty<*>, value: V) {
    this.put(property.name, value)
}

/**
 * Stores the value of the property for the given object in this mutable map.
* @param thisRef the object for which the value is requested (not used).
* @param property the metadata for the property, used to get the name of property and store the value associated with that name in the map.
* @param value the value to set.
*/
@kotlin.internal.InlineOnly
public inline operator fun <V> Map<in String, @Exact V>.getValue(thisRef: Any?, property: KProperty<*>): V1 =
    @Suppress("UNCHECKED_CAST") (getOrImplicitDefault(property.name) as V1)

/**
 * Stores the value of the property for the given object in this mutable map.
* @param thisRef the object for which the value is requested (not used).
* @param property the metadata for the property, used to get the name of property and store the value associated with that name in the map.
* @param value the value to set.
*/
@kotlin.internal.InlineOnly
public inline operator fun <V> Map<in String, @Exact V>.getValue(thisRef: Any?, property: KProperty<*>): V1 =
    @Suppress("UNCHECKED_CAST") (getOrImplicitDefault(property.name) as V1)

/**
 * Stores the value of the property for the given object in this mutable map.
* @param thisRef the object for which the value is requested (not used).
* @param property the metadata for the property, used to get the name of property and store the value associated with that name in the map.
* @param value the value to set.
*/
@kotlin.internal.InlineOnly
public inline operator fun <V> Map<in String, @Exact V>.getValue(thisRef: Any?, property: KProperty<*>): V1 =
    @Suppress("UNCHECKED_CAST") (getOrImplicitDefault(property.name) as V1)

/**
 * Stores the value of the property for the given object in this mutable map.
* @param thisRef the object for which the value is requested (not used).
* @param property the metadata for the property, used to get the name of property and store the value associated with that name in the map.
* @param value the value to set.
*/
@kotlin.internal.InlineOnly
public inline operator fun <V> Map<in String, @Exact V>.getValue(thisRef: Any?, property: KProperty<*>): V1 =
    @Suppress("UNCHECKED_CAST") (getOrImplicitDefault(property.name) as V1)

/**
 * Stores the value of the property for the given object in this mutable map.
* @param thisRef the object for which the value is requested (not used).
* @param property the metadata for the property, used to get the name of property and store the value associated with that name in the map.
* @param value the value to set.
*/
@kotlin.internal.InlineOnly
public inline operator fun <V> Map<in String, @Exact V>.getValue(thisRef: Any?, property: KProperty<*>): V1 =
    @Suppress("UNCHECKED_CAST") (getOrImplicitDefault(property.name) as V1)

/**
 * Stores the value of the property for the given object in this mutable map.
* @param thisRef the object for which the value is requested (not used).
* @param property the metadata for the property, used to get the name of property and store the value associated with that name in the map.
* @param value the value to set.
*/
@kotlin.internal.InlineOnly
public inline operator fun <V> Map<in String, @Exact V>.getValue(thisRef: Any?, property: KProperty<*>): V1 =
    @Suppress("UNCHECKED_CAST") (getOrImplicitDefault(property.name) as V1)

/**
 * Stores the value of the property for the given object in this mutable map.
* @param thisRef the object for which the value is requested (not used).
* @param property the metadata for the property, used to get the name of property and store the value associated with that name in the map.
* @param value the value to set.
*/
@kotlin.internal.InlineOnly
public inline operator fun <V> Map<in String, @Exact V>.getValue(thisRef: Any?, property: KProperty<*>): V1 =
    @Suppress("UNCHECKED_CAST") (getOrImplicitDefault(property.name) as V1)

/**
 * Stores the value of the property for the given object in this mutable map.
* @param thisRef the object for which the value is requested (not used).
* @param property the metadata for the property, used to get the name of property and store the value associated with that name in the map.
* @param value the value to set.
*/
@kotlin.internal.InlineOnly
public inline operator fun <V> Map<in String, @Exact V>.getValue(thisRef: Any?, property: KProperty<*>): V1 =
    @Suppress("UNCHECKED_CAST") (getOrImplicitDefault(property.name) as V1)
MutableMapWithDefault -> this.map.withDefault(defaultValue)
else -> MutableMapWithDefaultImpl(this, defaultValue)
})

private interface MapWithDefault<

private class MapWithDefaultImpl<K, out V>(public override val map: Map<K, V>, private val default: (key: K) -> V) : MapWithDefault<K, V>(public override val map: MutableMap<K, V>, private val default: (key: K) -> V) {

/*
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("CollectionsKt")

package kotlin.collections

import kotlin.random.Random

/**
 * Removes a single instance of the specified element from this
 * collection, if it is present.
 * @return `true` if the element has been successfully removed; `false` if it was not present in the collection.
 */
@kotlin.internal.InlineOnly
public inline fun <T> MutableCollection<out T>.remove(element: T): Boolean =
    @Suppress("UNCHECKED_CAST") (this as MutableCollection<T>).remove(element)

/**
 * Retains only the elements in this collection that are contained in the specified collection.
 * @return `true` if any element was removed from the collection, `false` if the collection was not modified.
 */
@kotlin.internal.InlineOnly
public inline fun <T> MutableCollection<out T>.retainAll(elements: Collection<T>): Boolean =
    @Suppress("UNCHECKED_CAST") (this as MutableCollection<T>).retainAll(elements)

/**
 * Adds the specified [element] to this mutable collection.
 */
@kotlin.internal.InlineOnly
operator fun <T> MutableCollection<in T>.plusAssign(element: T) =
    @Suppress("UNCHECKED_CAST") (this as MutableCollection<T>).plusAssign(element)
MutableCollection<in T>.plusAssign(elements: Array<T>) { this.addAll(elements) * Adds all elements of the given [elements] sequence to this mutable collection.

MutableCollection<in T>.plusAssign(elements: Sequence<T>) { this.addAll(elements) * Adds all elements of the given [elements] sequence to this mutable collection.

MutableCollection<in T>.plusAssign(element: T) { this.remove(element) * Removes all elements contained in the given [elements] collection from this mutable collection.

MutableCollection<in T>.minusAssign(elements: Iterable<T>) { this.removeAll(elements) * Removes all elements contained in the given [elements] sequence from this mutable collection.

MutableCollection<in T>.minusAssign(elements: Array<T>) { this.removeAll(elements) * Removes all elements contained in the given [elements] array from this mutable collection.

MutableCollection<in T>.minusAssign(elements: Sequence<T>) { this.removeAll(elements) * Removes all elements contained in the given [elements] sequence from this mutable collection.

MutableCollection<in T>.minusAssign(elements: Collection<T>) { this.removeAll(elements) * Removes all elements contained in the given [elements] collection from this mutable collection.

MutableCollection<in T>.minusAssign(elements: Set<T>) { this.removeAll(elements) * Removes all elements contained in the given [elements] set from this mutable collection.

MutableCollection<in T>.minusAssign(elements: MutableList<T>) { this.removeAll(elements) * Removes all elements contained in the given [elements] mutable list from this mutable collection.

MutableCollection<in T>.minusAssign(elements: DynamicArray<T>) { this.removeAll(elements) * Removes all elements contained in the given [elements] dynamic array from this mutable collection.

MutableCollection<in T>.minusAssign(elements: Array<T>) { this.removeAll(elements) * Removes all elements contained in the given [elements] array from this mutable collection.

MutableCollection<in T>.minusAssign(elements: Collection<in T>) { this.removeAll(elements) * Removes all elements contained in the given [elements] collection from this mutable collection.

MutableCollection<in T>.minusAssign(elements: Array<in T>) { this.removeAll(elements) * Removes all elements contained in the given [elements] array from this mutable collection.

MutableCollection<in T>.minusAssign(elements: Sequence<in T>) { this.removeAll(elements) * Removes all elements contained in the given [elements] sequence from this mutable collection.

MutableCollection<in T>.minusAssign(elements: Collection<in T>) { this.removeAll(elements) * Removes all elements contained in the given [elements] collection from this mutable collection.

MutableCollection<in T>.minusAssign(elements: Array<in T>) { this.removeAll(elements) * Removes all elements contained in the given [elements] array from this mutable collection.

MutableCollection<in T>.minusAssign(elements: Sequence<in T>) { this.removeAll(elements) * Removes all elements contained in the given [elements] sequence from this mutable collection.

MutableCollection<in T>.minusAssign(elements: Collection<in T>) { this.removeAll(elements) * Removes all elements contained in the given [elements] collection from this mutable collection.

MutableCollection<in T>.minusAssign(elements: Array<in T>) { this.removeAll(elements) * Removes all elements contained in the given [elements] array from this mutable collection.

MutableCollection<in T>.minusAssign(elements: Sequence<in T>) { this.removeAll(elements) * Removes all elements contained in the given [elements] sequence from this mutable collection.

MutableCollection<in T>.minusAssign(elements: Collection<in T>) { this.removeAll(elements) * Removes all elements contained in the given [elements] collection from this mutable collection.

MutableCollection<in T>.minusAssign(elements: Array<in T>) { this.removeAll(elements) * Removes all elements contained in the given [elements] array from this mutable collection.

MutableCollection<in T>.minusAssign(elements: Sequence<in T>) { this.removeAll(elements) * Removes all elements contained in the given [elements] sequence from this mutable collection.

MutableCollection<in T>.minusAssign(elements: Collection<in T>) { this.removeAll(elements) * Removes all elements contained in the given [elements] collection from this mutable collection.

MutableCollection<in T>.minusAssign(elements: Array<in T>) { this.removeAll(elements) * Removes all elements contained in the given [elements] array from this mutable collection.

MutableCollection<in T>.minusAssign(elements: Sequence<in T>) { this.removeAll(elements) * Removes all elements contained in the given [elements] sequence from this mutable collection.

MutableCollection<in T>.minusAssign(elements: Collection<in T>) { this.removeAll(elements) * Removes all elements contained in the given [elements] collection from this mutable collection.

MutableCollection<in T>.minusAssign(elements: Array<in T>) { this.removeAll(elements) * Removes all elements contained in the given [elements] array from this mutable collection.

MutableCollection<in T>.minusAssign(elements: Sequence<in T>) { this.removeAll(elements) * Removes all elements contained in the given [elements] sequence from this mutable collection.
predicateResultToRemove) {\n    remove()\n    result = true\n    }\n    return result\n}\n\n/**\n * Removes the element at the specified [index] from this list.\n * In Kotlin one should use the [MutableList.removeAt] function instead.\n * @param index\n * @return \n */\n\npublic fun <T> MutableList<T>.remove(index: Int): T = removeAt(index)\n\n/**\n * Removes the first element from this mutable list and returns that removed element, or throws [NoSuchElementException] if this list is empty.\n */\n\npublic fun <T> MutableList<T>.removeFirst(): T = if (isEmpty()) throw NoSuchElementException("List is empty.") else removeAt(0)\n\n/**\n * Removes the first element from this mutable list and returns that removed element, or returns `null` if this list is empty.\n */\n\npublic fun <T> MutableList<T>.removeFirstOrNull(): T? = if (isEmpty()) null else removeAt(0)\n\n/**\n * Removes all elements from this [MutableList] that match the given [predicate].\n * @param predicate\n * @return \n */\n\npublic fun <T> MutableList<T>.removeAll(predicate: (T) -> Boolean): Boolean = filterInPlace(predicate, true)\n\n/**\n * Retains only elements of this [MutableList] that match the given [predicate].\n * @param predicate\n * @return \n */\n\npublic fun <T> MutableList<T>.retainAll(predicate: (T) -> Boolean): Boolean = filterInPlace(predicate, false)\n\nprivate fun <T> MutableList<T>.filterInPlace(predicate: (T) -> Boolean, predicateResultToRemove: Boolean): Boolean {
    if (this !is RandomAccess)\n        return (this as MutableList<T>).filterInPlace(predicate, predicateResultToRemove)\n
    var writeIndex: Int = 0\n    for (readIndex in 0..lastIndex) {\n        val element = this[readIndex]\n        if (predicate(element) == predicateResultToRemove)\n            continue\n
        if (writeIndex != readIndex)\n            this[writeIndex] = element\n
        writeIndex++\n    }\n    if (writeIndex < size) {
        for (removeIndex in lastIndex downTo writeIndex)\n            removeAt(removeIndex)\n
        return true\n    } else {\n        return false\n    }
}

Open Source Used In security_automation identity-guest-cs

COPYRIGHT 2010-2018 Jetbrains s.r.o. and Kotlin Programming Language contributors. Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
/* public fun <T> List<T>.asReversed(): List<T> = ReversedListReadOnly(this) */
/* * Returns a reversed mutable view of the original mutable List.
* * All changes made in the original list will be reflected in the reversed
* one and vice versa. */

/* @sample samples.collections.ReversedViews
 */

/**
 * Returns a reversed
 * mutable view of the original mutable List.
 * * Builds a
 * [Sequence] lazily yielding values one by one.
 * * @see kotlin.sequences.generateSequence
 */

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming
 * Language contributors.
 * * Use of this source code is governed by the Apache 2.0 license that can
 * be found in the license/LICENSE.txt file.
 */

/*
 * package kotlin.sequences
 */

@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("SequencesKt")

import kotlin.coroutines.*
import kotlin.coroutines.intrinsics.*
import kotlin.experimental.ExperimentalTypeInference

/**
 * Builds an [Iterator] lazily yielding values one by one.
 * * @see sample.collections.Sequences
 */

/**
 * The scope for yielding values of a [Sequence] or an [Iterator],
 * provides [yield] and [yieldAll] suspension functions.
 * * @sample samples.collections.Sequences
 */

private typealias State = Int

private const val State_NotReady: State = 0
private const val
private fun partition(
    array: UShortArray, left: Int, right: Int): Int {
    var i = left
    var j = right
    val pivot = array[(left + right) / 2]
    while (i <= j) {
        while (array[i] < pivot) i++
        while (array[j] > pivot) j--
        if (i <= j) {
            val tmp = array[i]
            array[i] = array[j]
            array[j] = tmp
            i++
            j--
        }
    }
    return i
}

@Suppress("UNCHECKED_CAST")
private fun quickSort(array: UByteArray, left: Int, right: Int) {
    val index = partition(array, left, right)
    if (index < right)
        quickSort(array, index, right)
}

// UByteArray
val start = startIndex
    val end = start.forward(n)

    if (start > end) {
        buffer.fill(null, start, end)
    } else {
        @Suppress("UNCHECKED_CAST")
        setNext(buffer[index] as T)
    }

    override fun computeNext() {
        if (count == 0) {
            done()
            return
        } else {
            @Suppress("UNCHECKED_CAST")
            setNext(buffer[index].forward(1) as T)
            index = index.forward(1)
            count--
        }
    }

    fun isFull() = false

    constructor(capacity: Int) : this(arrayOfNulls<Any?>(capacity), 0)

    @Suppress("NOTHING_TO_INLINE")
    private fun buffer[startIndex: Int, size: Int] result: Array<T?> =
        if (array.size < this.size) array.copyOf(this.size) else array as Array<T?>

    val fromIndex: Int
    val capacity: Int
    val startIndex: Int

    override var size: Int = filledSize
    override var count: Int = 0

    override fun get(index: Int): E {
        checkElementIndex(index, _size)
        checkRangeIndexes(fromIndex, toIndex, list.size)
        return list[fromIndex + index]
    }

    override fun set(index: Int, element: E) {
        checkElementIndex(index, _size)
        checkRangeIndexes(fromIndex, toIndex, list.size)
        list[fromIndex + index] = element
    }

    override fun remove(): E {
        checkElementIndex(index, _size)
        return removeLast()
    }

    override fun removeFirst(): E {
        require(n >= 0) { "n shouldn't be negative but it is $n" }
        require(n <= size) { "n shouldn't be greater than the buffer size: n = $n, size = $size" }
        if (n > 0) {
            val start = startIndex
            val end = start.forward(n)
            if (start > end) {
                buffer.fill(null, start, end)
                start = end
                size = size - n
            } else {
                buffer.fill(null, start, end)
            }
        }

        return @ExperimentalUnsignedTypes
private fun partition(
    array: UByteArray, left: Int, right: Int): Int {
    var i = left
    var j = right
    val pivot = array[(left + right) / 2]
    while (i <= j) {
        while (array[i] < pivot) i++
        while (array[j] > pivot) j--
        if (i <= j) {
            val tmp = array[i]
            array[i] = array[j]
            array[j] = tmp
            i++
            j--
        }
    }
    return i
}

@Suppress("UNCHECKED_CAST")
private fun quickSort(array: UByteArray, left: Int, right: Int) {
    val index = partition(array, left, right)
    if (index < right)
        quickSort(array, index, right)
}

@Suppress("UNCHECKED_CAST")
private fun quickSort(array: UShortArray, left: Int, right: Int) {
    val index = partition(array, left, right)
    if (index < right)
        quickSort(array, index, right)
}

@Suppress("UNCHECKED_CAST")
private fun quickSort(array: UShortArray, left: Int, right: Int) {
    val index = partition(array, left, right)
    if (index < right)
        quickSort(array, index, right)
}
The code provided contains implementations of the quick sort algorithm for different numeric types: `UShortArray`, `UIntArray`, and `ULongArray`. The algorithm uses a partitioning technique to sort arrays in-place. It includes the partition function and the quick sort function that recursively calls itself on the sub-arrays.

At the bottom of the code, there is also the definition of a `ContractBuilder` interface and an `ExperimentalContracts` annotation. The `ContractBuilder` interface allows describing contracts of user functions, and `ExperimentalContracts` is used to opt-in for the experimental contract declaration API. The `sortArray` function is used to sort arrays of various numeric types, and it leverages the quick sort algorithm for this purpose.

The code also includes a note about copyright and the Apache 2.0 license, indicating that this source code is governed by the Apache 2.0 license that can be found in the LICENSE.txt file. Additionally, there is a note about the ExperimentalContracts annotation, which must be accepted either directly or by using the compiler argument `-Xopt-in=kotlin.contracts.ExperimentalContracts`.
conditional effect that happens in such case.

```kotlin
*/
/**
 * Describes a situation when a function returns normally with any value that is not `null`.
 * Use [SimpleEffect.implies] function to describe a conditional effect that happens in such case.
 */

// @sample samples.contracts.returnsFalseContract
@ContractsDsl public fun returns(value: Any?): Returns

// @sample samples.contracts.returnsNullContract
/**
 * Describes a situation when a function
 * returns null.
 *
 */

// @sample samples.contracts.returnsNotNullContract
@ContractsDsl public fun returnsNotNull(): ReturnsNotNull

/**
 * Specifies that the function parameter [lambda] is invoked in place.
 *
 * This contract specifies:
 * 1. the function [lambda] can only be invoked during the call of the owner function,
 * and it won't be invoked after that owner function call is completed;
 * 2. (optionally) the function [lambda] is invoked the amount of times specified by the
 * [kind] parameter;
 * see the [InvocationKind] enum for possible values.
 *
 * A function declaring the `callsInPlace` effect must be _inline_.
 */

// @sample samples.contracts.callsInPlaceAtMostOnceContract
@ContractsDsl public fun <R> callsInPlace(lambda: Function<R>, kind: InvocationKind = InvocationKind.UNKNOWN): CallsInPlace

/**
 * Specifies how many times a function invokes its function parameter in place.
 *
 */
@ContractsDsl
@ExperimentalContracts
@SinceKotlin("1.3")
public enum class InvocationKind {
    AT_MOST_ONCE,
    AT_LEAST_ONCE,
    EXACTLY_ONCE,
    UNKNOWN
}

/**
 * Specifies the contract of a function.
 *
 * The contract description must be at the beginning of a function and have at least one
 * effect.
 * Only the top-level functions can have a contract for now.
 *
 */
@ContractsDsl
@ExperimentalContracts
@InlineOnly
@SinceKotlin("1.3")
@Suppress("UNUSED_PARAMETER")
public inline fun contract(builder: ContractBuilder.() -> Unit) {

@
```
The key that defines the context interceptor. *

A companion object Key:

```kotlin
Corouti
```
@Suppress("UNCHECKED_CAST")
    if (this.key == key) this as E else null

    override fun <R> fold(initial: R, operation: (R, Element) -> R): R
        = operation(initial, this)

    override fun minusKey(key: Key<*>): CoroutineContext =
        if (this.key == key) EmptyCoroutineContext else this

    /*
     * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
     * Use of this source code is governed by the Apache 2.0 license that can be found in the
     * license/LICENSE.txt file.
     */

    package kotlin.coroutines

    import kotlin.coroutines.CoroutineContext.Element

    /**
     * Base class for [CoroutineContext.Element] implementations.
     */
    @SinceKotlin("1.3")
    public abstract class AbstractCoroutineContextElement(public override val key: Key<*>)
        : Element

    /**
     * Base class for [CoroutineContext.Key] associated with polymorphic
     * [CoroutineContext.Element] implementation.
     * Polymorphic element implementation implies delegating its
     * [get][Element.get] and [minusKey][Element.minusKey] to [getPolymorphicElement] and
     * [minusPolymorphicKey] respectively.
     */
    @SinceKotlin("1.3")
    @ExperimentalStdlibApi
    public abstract class AbstractCoroutineContextKey<B : Element, E : B>(
        baseKey: Key<B>,
        private val safeCast: (element: Element) -> E?
    ) : Key<E> {
        private val topmostKey: Key<*> = if (baseKey is AbstractCoroutineContextKey<*, *>)
            baseKey.topmostKey else baseKey

        internal fun tryCast(element: Element): E? = safeCast(element)

        internal fun isSubKey(key: Key<*>): Boolean = key === this ||
            topmostKey === key

        @SinceKotlin("1.3")
        @ExperimentalStdlibApi
        public abstract fun <E : Element> Element.getPolymorphicElement(key: Key<E>): E?

        @SinceKotlin("1.3")
        @ExperimentalStdlibApi
        public abstract fun Element.minusPolymorphicKey(key: Key<*>): CoroutineContext

        /**
         * An empty coroutine context.
         */
        @SinceKotlin("1.3")
        public object EmptyCoroutineContext : CoroutineContext, Serializable
            [public override fun readResolve(): Any = EmptyCoroutineContext]

        @SinceKotlin("1.3")
        @ExperimentalStdlibApi
        public fun Element.minusPolymorphicKey(key: Key<*>): CoroutineContext
            = if (key is AbstractCoroutineContextKey<*, *> &
                    key.tryCast(this) != null) EmptyCoroutineContext else this

        @SinceKotlin("1.3")
        @ExperimentalStdlibApi
        public object EmptyCoroutineContext : CoroutineContext, Serializable
Key<E>): E? = null
    public override fun <R> fold(initial: R, operation: (R, Element) -> R): R = initial
    public override fun <R> fold(initial: R, operation: (R, Element) -> R): R = initial
    public override fun plus(context: CoroutineContext): CoroutineContext = context
    public override fun minusKey(key: Key<*>): CoroutineContext = this
    public override fun hashCode(): Int = 0
    public override fun toString(): String = "EmptyCoroutineContext"
}

//--------------------- internal impl ---------------------

// this class is not exposed, but is hidden inside implementations
// this is a left-biased list, so that `plus` works naturally
@SinceKotlin("1.3")
internal class CombinedContext(
    private val left: CoroutineContext,
    private val element: Element
) : CoroutineContext, Serializable {

    override fun <E : Element> get(key: Key<E>): E? = element[key] ?: let { return it } val next = cur.left if (next is CombinedContext) { cur = cur.next = next } else { cur = next } return EmptyCoroutineContext

    override fun <R> fold(initial: R, operation: (R, Element) -> R): R = operation(left.fold(initial, operation), element)

    override fun minusKey(key: Key<*>): CoroutineContext {
        element[key] ?.let { return left } val newLeft = left.minusKey(key)
        return when {
            newLeft === left -> this
            newLeft === EmptyCoroutineContext -> element
            else -> CombinedContext(newLeft, element)
        }
    }

    private fun size(): Int {
        var cur = this
        var size = 2
        while (true) {
            cur = cur.left as? CombinedContext ?: return size
            size++
        }
    }

    private fun contains(element: Element): Boolean = get(element.key) == element

    private fun containsAll(context: CombinedContext): Boolean {
        var cur = context
        while (true) {
            if (!contains(cur.element)) return false
            val next = cur.left
            if (next is CombinedContext) {
                cur = next
            } else {
                return contains(next as Element)
            }
        }
    }

    override fun equals(other: Any?): Boolean = this === other || other is CombinedContext && other.size() == size() && other.containsAll(this)

    override fun hashCode(): Int = left.hashCode() + element.hashCode()

    override fun toString(): String = "[$" + fold("") { acc, element ->
        if (acc.isEmpty()) element.toString() else "$acc, $element"
    } + "]"

    private fun writeReplace(): Any {
        val n = size()
        val elements = Array(n) { 0 } as Array<CoroutineContext>
        fold(Unit) { _, element -> elements[index++] = element }
        check(index == n)
        @Suppress("UNCHECKED_CAST")
        return Serialized(elements)
    }

    private class Serialized(val elements: Array<CoroutineContext>) : Serializable {
        companion object {
            private const val serialVersionUID: Long = 0L
        }

        private fun readResolve(): Any = elements.fold(EmptyCoroutineContext, CoroutineContext::plus)
    }

    internal.InlineOnly
    public override fun <E> get(key: Key<E>): E? = null
    public override fun <R> fold(initial: R, operation: (R, Element) -> R): R = initial
    public override fun <R> fold(initial: R, operation: (R, Element) -> R): R = initial
    public override fun plus(context: CoroutineContext): CoroutineContext = context
    public override fun minusKey(key: Key<*>): CoroutineContext = this
    public override fun hashCode(): Int = 0
    public override fun toString(): String = "EmptyCoroutineContext"
}

@file:kotlin.jvm.JvmName("IntrinsicsKt")
@file:kotlin.jvm.JvmMultifileClass
package kotlin.coroutines.intrinsics

import kotlin.contracts.*
import kotlin.coroutines.
import kotlin.internal.InlineOnly

/**
 * Obtains the current continuation instance inside suspend functions and either
 * suspends currently running coroutine or returns result immediately without suspension.
 * If the [block] returns the special [COROUTINE_SUSPENDED] value, it means that suspend function did suspend the execution
 * and will not return any result immediately. In this case, the [Continuation] provided to the [block] shall be
 * resumed by invoking [Continuation.resumeWith] at some moment in the future when the result becomes
 * available to resume the computation. Otherwise, the return value of the [block] must have a type assignable
 * to [T] and represents the result of this suspend function.
 * That means that the execution was not suspended and the [Continuation] provided to the [block] shall not be invoked.
 * As the result type of the [block] is declared as Any, and cannot be correctly type-checked, its proper return type remains on the conscience of the suspend function's author. [ Invocation of [Continuation.resumeWith] resumes coroutine directly in the invoker's thread without going through the
 * [ContinuationInterceptor] that might be present in the coroutine's [CoroutineContext]. It is the invoker's responsibility to ensure that a proper invocation context is established. [ Note that it is not recommended to call either [Continuation.resume] nor [Continuation.resumeWithException] functions
 * synchronously in the same stackframe where suspension function is run. Use [suspendCoroutine] as a safer way to obtain current instance.
 */
suspendCoroutineUninterceptedOrReturn(crossinline block: (Continuation<T>) -> Any?): T {
    contract {
        callsInPlace(block, InvocationKind.EXACTLY_ONCE)
    }
    throw NotImplementedError("Implementation of suspendCoroutineUninterceptedOrReturn is intrinsic")
}

/**
 * This value is used as a return value of `suspendCoroutineUninterceptedOrReturn` `block` argument to state that execution was suspended and will not return any result immediately.
 * **Note: this value should not be used in general code.** Using it outside of the context of `suspendCoroutineUninterceptedOrReturn` function return value (including, but not limited to, storing this value in other properties, returning it from other functions, etc) can lead to unspecified behavior of the code.
 */
// It is implemented as property with getter to avoid ProGuard <clinit> problem with multifile IntrinsicsKt class
@SinceKotlin("1.3")
public val COROUTINE_SUSPENDED: Any get() = CoroutineSingletons.COROUTINE_SUSPENDED

// Using enum here ensures two important properties:
// 1. It makes SafeContinuation serializable with all kinds of serialization frameworks (since all of them natively support enums)
// 2. It improves debugging experience, since you clearly see toString() value of those objects and what package they come from
@SinceKotlin("1.3")
@PublishedApi // This class is Published API via serialized representation of SafeContinuation, don't rename/move internal enum class
internal enum class CoroutineSingletons {
    COROUTINE_SUSPENDED, UNDECIDED, RESUMED
}

/** Performs a bitwise AND operation between the two values. */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public inline infix fun Byte.and(other: Byte): Byte = (this.toInt() and other.toInt()).toByte()

/** Performs a bitwise OR operation between the two values. */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public inline infix fun Byte.or(other: Byte): Byte = (this.toInt() or other.toInt()).toByte()

/** Performs a bitwise XOR operation between the two values. */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public inline infix fun Byte.xor(other: Byte): Byte = (this.toInt() xor other.toInt()).toByte()

/** Inverts the bits in this value. */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public inline fun Byte.inv(): Byte = (this.toInt().inv()).toByte()

/** Performs a bitwise AND operation between the two values. */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public inline infix fun Short.and(other: Short): Short = (this.toInt() and other.toInt()).toShort()

/** Performs a bitwise OR operation between the two values. */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public inline infix fun Short.or(other: Short): Short = (this.toInt() or other.toInt()).toShort()

/** Performs a bitwise XOR operation between the two values. */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public inline infix fun Short.xor(other: Short): Short = (this.toInt() xor other.toInt()).toShort()

/** Inverts the bits in this value. */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public inline fun Short.inv(): Short = (this.toInt().inv()).toShort()
*\n*\n*\n*\n
*/
@Target(AnnotationTarget.TYPE)
@Retention(AnnotationRetention.BINARY)
internal annotation class NoInfer

/**
 * Specifies that the constraint built for the type during type inference should be an equality one.
 */
@Target(AnnotationTarget.TYPE)
@Retention(AnnotationRetention.BINARY)
internal annotation class Exact

/**
 * Specifies that a corresponding member has the lowest priority in overload resolution.
 */
@Target(AnnotationTarget.FUNCTION,
 AnnotationTarget.PROPERTY)
@Retention(AnnotationRetention.BINARY)
internal annotation class LowPriorityInOverloadResolution

/**
 * Specifies that the corresponding member has the highest priority in overload resolution. Effectively this means that
 * an extension annotated with this annotation will win in overload resolution over a member with the same signature.
 */
@Target(AnnotationTarget.FUNCTION,
 AnnotationTarget.PROPERTY)
@Retention(AnnotationRetention.BINARY)
internal annotation class HidesMembers

/**
 * The value of this type parameter should be mentioned in input types (argument types, receiver type or expected type).
 */
@Target(AnnotationTarget.FUNCTION,
 AnnotationTarget.PROPERTY,
 AnnotationTarget.PROPERTY_GETTER,
 AnnotationTarget.PROPERTY_SETTER)
@Retention(AnnotationRetention.BINARY)
internal annotation class OnlyInputTypes

/**
 * Specifies that this function should not be called directly without inlining.
 */
@Target(AnnotationTarget.FUNCTION,
 AnnotationTarget.PROPERTY,
 AnnotationTarget.PROPERTY_GETTER,
 AnnotationTarget.PROPERTY_SETTER)
@Retention(AnnotationRetention.BINARY)
internal annotation class InlineOnly

/**
 * Specifies that this declaration can have dynamic receiver type.
 */
@Target(AnnotationTarget.FUNCTION,
 AnnotationTarget.PROPERTY)
@Retention(AnnotationRetention.BINARY)
internal annotation class DynamicExtension

/**
 * The value of this parameter should be a property reference expression (`this::foo`), referencing a `lateinit` property.
 * The backing field of which is accessible at the point where the corresponding argument is passed.
 */
@Target(AnnotationTarget.VALUE_PARAMETER)
@Retention(AnnotationRetention.BINARY)
@SinceKotlin("1.2")
internal annotation class AccessibleLateinitPropertyLiteral

/**
 * Specifies that this declaration is only completely supported since the specified version.
 * The Kotlin compiler of an earlier version is going to report a diagnostic on usages of this declaration.
 * The diagnostic message can be specified with [message], or via [errorCode] (takes less space, but might not be immediately clear to the user). The diagnostic severity can be specified with [level]: WARNING/ERROR mean that either a warning or an error is going to be reported, HIDDEN means that the declaration is going to be removed from resolution completely. [versionKind] specifies which version should be compared with the [version] value, when compiling the usage of the annotated declaration. Note that prior to 1.2, only [RequireKotlinVersionKind.LANGUAGE_VERSION] was supported, so the Kotlin compiler before 1.2 is going to treat any [RequireKotlin] as if it requires the language version. Since 1.2, the Kotlin compiler supports [versionKind] * [RequireKotlinVersionKind.LANGUAGE_VERSION], [RequireKotlinVersionKind.COMpiler_VERSION] and [RequireKotlinVersionKind.API_VERSION]. If the actual value of [versionKind] is something different (e.g. a new version kind, added in future versions of Kotlin), Kotlin 1.2 is going to ignore this [RequireKotlin] altogether, where as Kotlin before 1.2 is going to treat this as a requirement on the language version. This annotation is erased at compile time; its arguments are stored in a more compact form in the Kotlin metadata.
 */
@Target(AnnotationTarget.CLASS,
 AnnotationTarget.FUNCTION,
 AnnotationTarget.PROPERTY,
 AnnotationTarget.CONSTRUCTOR,
 AnnotationTarget.TYPEALIAS)
@Retention(AnnotationRetention.SOURCE)
@Repeatable
@SinceKotlin("1.2")
internal annotation class RequireKotlin

/**
 * The kind of the version that is required by [RequireKotlin].
 */
@SinceKotlin("1.2")
internal enum class RequireKotlinVersionKind {
    LANGUAGE_VERSION,
    COMPILER_VERSION,
    API_VERSION,
}

/**
 * Specifies that this declaration is a part of special DSL, used for constructing function's contract.
 */
@Target(AnnotationTarget.FUNCTION,
 AnnotationTarget.PROPERTY)
@Retention(AnnotationRetention.BINARY)
internal annotation class OpenSourceUsedIn security_automation identity-guest-cs 1771
class ContractsDsl

Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

*package kotlin.properties

import kotlin.reflect.KProperty

*public object Delegates {

*Returns a property delegate for a read/write property with a non-null value that is initialized not during object construction time but at a later time.

*assigned results in an exception.

*assigned

*name: "ContractsDsl"

*/

package kotlin.properties

import kotlin.reflect.KProperty

/**
 * Standard property delegates.
 */

public object Delegates {

*public fun T : Any? notNull():

ReadWriteProperty<T : Any?, T> =NotNullVar()<

*returns a property delegate for a read/write property that calls a specified callback function when changed.

*param initialValue the initial value of the property.

*param onChange the callback which is called after the change of the property is made. The value of the property has already been changed when this callback is invoked.

*param samples.properties.Delegates.observableDelegate

*public inline fun T observable(initialValue: T, crossinline onChange: (property: KProperty<*>, oldValue: T, newValue: T) -> Unit):

object : ObservableProperty<T>(initialValue) {

override fun afterChange(property: KProperty<*>, oldValue: T, newValue: T) = onChange(property, oldValue, newValue)

}

private class NotNullVar<T : Any>() : ReadWriteProperty<Any?, T> {

private var value: T? = null

override fun getValue(thisRef: Any?, property: KProperty<*>): T {

return value ?: throw IllegalStateException("Property ${property.name} should be initialized before get.")

}

override fun setValue(thisRef: Any?, property: KProperty<*>, value: T) {

this.value = value

}

}

/*
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

*/

package kotlin.properties

import kotlin.reflect.KProperty

/**
 * Base interface that can be used for implementing property delegates of read-only properties.
 *
 * This is provided only for convenience; you don't have to extend this interface as long as your property delegate has methods with the same signatures:

*param T the type of object which owns the delegated property.

*param V the type of the property value.

*public fun interface ReadOnlyProperty<T : Any?, T> out V>

override fun getValue(thisRef: Any?, property: KProperty<*>): V

override fun setValue(thisRef: Any?, property: KProperty<*>): V

*/

/*
 * Copyright 2010-2020 Jetbrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

*/

package kotlin.properties

import kotlin.reflect.KProperty

/**
 * Base interface that can be used for implementing property delegates of read-write properties.
 *
 * This is provided only for convenience; you don't have to extend this interface as long as your property delegate has methods with the same signatures:

*param T the type of object which owns the delegated property.

*param V the type of the property value.

*/

/*
 * Copyright 2010-2020 Jetbrains s.r.o. and Kotlin Programming Language contributors.
 */

package kotlin.properties

import kotlin.reflect.KProperty

/**
 * Base interface that can be used for implementing property delegates of read-write properties.
 *
 * This is provided only for convenience; you don't have to extend this interface as long as your property delegate has methods with the same signatures:

*param T the type of object which owns the delegated property.

*param V the type of the property value.

*/

/*
 * Copyright 2010-2020 Jetbrains s.r.o. and Kotlin Programming Language contributors.
 */

package kotlin.properties

import kotlin.reflect.KProperty

/**
 * Base interface that can be used for implementing property delegates of read-write properties.
 *
 * This is provided only for convenience; you don't have to extend this interface as long as your property delegate has methods with the same signatures:

*param T the type of object which owns the delegated property.

*param V the type of the property value.

*/

/*
 * Copyright 2010-2020 Jetbrains s.r.o. and Kotlin Programming Language contributors.
 */

package kotlin.properties

import kotlin.reflect.KProperty

/**
 * Base interface that can be used for implementing property delegates of read-write properties.
 *
 * This is provided only for convenience; you don't have to extend this interface as long as your property delegate has methods with the same signatures:

*param T the type of object which owns the delegated property.

*param V the type of the property value.

*/

/*
 * Copyright 2010-2020 Jetbrains s.r.o. and Kotlin Programming Language contributors.
 */
the property.

 * @param value the value to set.
 */
 public operator fun setValue(thisRef: T, property: KProperty<*>, value: V) {
     return super.setValue(thisRef, property, value)
 }

 /**
 * Base interface that can be used for implementing property delegate providers.
 * This is provided only for convenience; you don't have to extend this interface as long as you
 * delegate provider has a method with the same signature.
 */
 @param T the type of object which owns the delegated property.
 @param D the type of property delegates this provider provides.
 */
@SinceKotlin("1.4")
public fun interface PropertyDelegateProvider<T: Any, D: Any> {
    /**
     * Returns the delegate of the property for the given object.
     * This function can be used to extend the logic of creating the object (e.g. perform validation checks) to which the property implementation is delegated.
     * @param thisRef the object for which property delegate is requested.
     * @param property the metadata for the property.
     * @return the property delegate.
     */
    public operator fun provideDelegate(thisRef: T, property: KProperty<*>): D
}

/**
 * Base interface that can be used for implementing property delegate providers.
 *
 * This is provided only for convenience; you don't have to extend this interface as long as you
 * delegate provider has a method with the same signature.
 */
@param T the type of object which owns the delegated property.
 @param D the type of property delegates this provider provides.
 */
@SinceKotlin("1.4")
public fun interface PropertyDelegateProvider<T: Any, D: Any> {
    /**
     * Returns the delegate of the property for the given object.
     * This function can be used to extend the logic of creating the object (e.g. perform validation checks) to which the property implementation is delegated.
     * @param thisRef the object for which property delegate is requested.
     * @param property the metadata for the property.
     * @return the property delegate.
     */
    public operator fun provideDelegate(thisRef: T, property: KProperty<*>): D
}

/**
 * Base interface that can be used for implementing property delegate providers.
 *
 * This is provided only for convenience; you don't have to extend this interface as long as you
 * delegate provider has a method with the same signature.
 */
@param T the type of object which owns the delegated property.
 @param D the type of property delegates this provider provides.
 */
@SinceKotlin("1.4")
public fun interface PropertyDelegateProvider<T: Any, D: Any> {
    /**
     * Returns the delegate of the property for the given object.
     * This function can be used to extend the logic of creating the object (e.g. perform validation checks) to which the property implementation is delegated.
     * @param thisRef the object for which property delegate is requested.
     * @param property the metadata for the property.
     * @return the property delegate.
     */
    public operator fun provideDelegate(thisRef: T, property: KProperty<*>): D
}
reference to a read-only or mutable property of type \([V]\) or its subtype.\n* \n* The reference has an unbound receiver of
type \([T]\).\n* \n* An extension operator that allows delegating a mutable member or extension property of type \([V]\) to a property reference to a member
or extension mutable property of the same type \([V]\).\n* \n* Example:\n* \n* ```
class Login(val username: String)
val Login.user by
Login::username
// equivalent to
val Login.user get() = this.username
```
range.

    *in * Generates a `Long` random value uniformly distributed between the specified `from` (inclusive)
    and `until` (exclusive) bounds.
    *in * @throws IllegalArgumentException if `from` is greater than or equal to
    `until`.
    *in * @sample samples.random.Randoms.nextLongFromUntil
    */

    public open fun nextLong(from: Long, until: Long): Long {
        checkRangeBounds(from, until)
        val n = until - from
        if (n > 0) {
            val nLow = toLong(n)
            val nHigh = (n shr 32).toInt()
            var rnd: Long
                = when {
                    (nLow != 0) -> {
                        val bitCount = fastLog2(nLow) + // toLong()
                        // toLong()
                        nextBits(bitCount).toInt()}
                        & 0xFFFF_FFFF
                    } else {
                        val bitCount = fastLog2(nLow) + // toLong()
                        // toLong()
                        nextBits(bitCount).toInt() + 32 + nextInt()
                        & 0xFFFF_FFFF
                    }
                }
            } else {
                var v: Long
                    do {
                        val bits = nextLong().ushr(1)
                        v = bits % n
                    } while (bits - v + (n - 1) < 0)
            } else {
                var v: Long
                    do {
                        val bits = nextLong().ushr(1)
                        v = bits % n
                    } while (true)
            }
        } else {
            if (n in from until until) return rnd
        }
        return from + nextDouble() * size
    } else {
        return if (r >= until) until.nextDown() else r
    }
}

    /**
     * Gets the next random `Double` value.
     *
     * @sample samples.random.Randoms.nextDouble
     */
    public open fun nextDouble(): Double = nextBits(26) / (1 shl

    /**
     * Gets the next random `Double` value.
     *
     * @return 
     */
    public open fun nextDouble(from: Double, until: Double): Double {
        checkRangeBounds(from, until)
        val size = until - from
        val r = if (size.isInfinite() && from.isFinite() && until.isFinite()) {
            val r1 = nextDouble() * (until / 2 - from / 2)
            from + r1 + r1
        } else {
            from + nextDouble() * size
        }

        val steps = (toIndex - fromIndex) / 4
        var position = fromIndex
        0..${array.size}." }
        require(fromIndex <= toIndex) { "fromIndex ($fromIndex) or toIndex ($toIndex) are out of range:
        " }

        val remainder = toIndex - position
        val vr = if (n > 0) {
            val nLow = toInt(n)
            val nHigh = (n shr 32).toInt()
            rnd = when {
                (nLow != 0) -> {
                    val bitCount = fastLog2(nLow) + // toLong()
                    // toLong()
                    nextBits(bitCount).toInt()}
                } else {
                    var v: Long
                        do {
                            val bits = nextLong().ushr(1)
                            v = bits % n
                        } while (true)
                }
            }
            return if (rnd in from until until) return rnd
        }
        return from + nextDouble() * size
    }

    /**
     * Gets the next random `Double` value
     *
     * @sample samples.random.Randoms.nextDouble
     */
    public open fun nextDoubleFromParts(nextBits(26), nextBits(27))

    /**
     * Gets the next random `Double` value
     *
     * @sample samples.random.Randoms.nextDouble
     */
    public open fun nextDouble(): Double = nextBits(26) / (1 shl

    /**
     * Gets the next random `Double` value
     *
     * @sample samples.random.Randoms.nextDouble
     */
    public open fun nextDouble(from: Double, until: Double): Double {
        checkRangeBounds(from, until)
        val size = until - from
        val r = if (size.isInfinite() && from.isFinite() && until.isFinite()) {
            val r1 = nextDouble() * (until / 2 - from / 2)
            from + r1 + r1
        } else {
            from + nextDouble() * size
        }

        val steps = (toIndex - fromIndex) / 4
        var position = fromIndex
        0..${array.size}." }
        require(fromIndex <= toIndex) { "fromIndex ($fromIndex) or toIndex ($toIndex) are out of range:
        " }

        val remainder = toIndex - position
        val vr = if (n > 0) {
            val nLow = toInt(n)
            val nHigh = (n shr 32).toInt()
            rnd = when {
                (nLow != 0) -> {
                    val bitCount = fastLog2(nLow) + // toLong()
                    // toLong()
                    nextBits(bitCount).toInt()}
                } else {
                    var v: Long
                        do {
                            val bits = nextLong().ushr(1)
                            v = bits % n
                        } while (true)
                }
            }
            return if (rnd in from until until) return rnd
        }
        return from + nextDouble() * size
   
nextBytes(ByteArray(size))

/**
 * The default random number generator.
 * On JVM this generator is thread-safe, its methods can be invoked from multiple threads.
 * @sample samples.random.Randoms.defaultRandom
 */

companion object Default : Random() {
    override fun nextInt(): Int = defaultRandom.nextInt()
    override fun nextLong(): Long = defaultRandom.nextLong()
    override fun nextDouble(): Double = defaultRandom.nextDouble()
    override fun nextFloat(): Float = defaultRandom.nextFloat()
    override fun nextBytes(array: ByteArray): ByteArray = defaultRandom.nextBytes(array)
    override fun nextBytes(size: Int): ByteArray = defaultRandom.nextBytes(size)
    override fun nextBytes(array: ByteArray, fromIndex: Int, toIndex: Int): ByteArray = defaultRandom.nextBytes(array, fromIndex, toIndex)
}

/**
 * Returns a repeatable random number generator seeded with the given [seed] `Int` value.
 * Two generators with the same seed produce the same sequence of values within the same version of Kotlin runtime.
 * Note: Future versions of Kotlin may change the algorithm of this seeded number generator so that it will return a sequence of values different from the current one for a given seed.
 * On JVM the returned generator is NOT thread-safe. Do not invoke it from multiple threads without proper synchronization.
 * @sample samples.random.Randoms.seededRandom
 */
@SinceKotlin("1.3")
public fun Random(seed: Int): Random = XorWowRandom(seed, seed.shr(31))

/**
 * Returns a repeatable random number generator seeded with the given [seed] `Long` value.
 * Two generators with the same seed produce the same sequence of values within the same version of Kotlin runtime.
 * Note: Future versions of Kotlin may change the algorithm of this seeded number generator so that it will return a sequence of values different from the current one for a given seed.
 * On JVM the returned generator is NOT thread-safe. Do not invoke it from multiple threads without proper synchronization.
 * @sample samples.random.Randoms.seededRandom
 */
@SinceKotlin("1.3")
public fun Random(seed: Long): Random = XorWowRandom(seed.toInt(), seed.shr(32).toInt())

/**
 * Gets the next random `Int` from the random number generator in the specified [range].
 * Generates an `Int` random value uniformly distributed in the specified [range]:
 * from `range.start` inclusive to `range.endInclusive` inclusive.
 * @throws IllegalArgumentException if [range] is empty.
 */
@SinceKotlin("1.3")
public fun Random.nextInt(range: IntRange): Int = when {
    range.isEmpty() -> throw IllegalArgumentException("Cannot get random in empty range: $range")
    range.last < Int.MAX_VALUE -> nextInt(range.first, range.last + 1)
    else -> nextInt()
}

/**
 * Gets the next random `Long` from the random number generator in the specified [range].
 * Generates a `Long` random value uniformly distributed in the specified [range]:
 * from `range.start` inclusive to `range.endInclusive` inclusive.
 * @throws IllegalArgumentException if [range] is empty.
 */
@SinceKotlin("1.3")
public fun Random.nextLong(range: LongRange): Long = when {
    range.isEmpty() -> throw IllegalArgumentException("Cannot get random in empty range: $range")
    range.last < Long.MAX_VALUE -> nextLong(range.first, range.last + 1)
    else -> nextLong()
}

internal expect fun defaultPlatformRandom(): Random
internal expect fun doubleFromParts(hi26: Int, low27: Int): Double

@OptIn(ExperimentalStdlibApi::class)
internal fun fastLog2(value: Int): Int = 31 - value.countLeadingZeroBits()

/** Takes upper [bitCount] bits (0..32) from this number. */
internal fun Int.takeUpperBits(bitCount: Int): Int =
    this.ushr(32 - bitCount) and (-bitCount).shr(31)

internal fun checkRangeBounds(from: Int, until: Int) = require(until > from) {
    boundsErrorMessage(from, until)
}

internal fun checkRangeBounds(from: Long, until: Long) = require(until > from) {
    boundsErrorMessage(from, until)
}

internal fun checkRangeBounds(from: Double, until: Double) =
require(until > from) { boundsErrorMessage(from, until) }

internal fun boundsErrorMessage(from: Any, until: Any) = "Random range is empty: [$from, $until)."

package kotlin.random

/**
 * Gets the next random [UInt] random value uniformly distributed between
 * [UInt.MIN_VALUE] and [UInt.MAX_VALUE] (inclusive).
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Random.nextUInt(): UInt = nextInt(0u, until = Int.MAX_VALUE) xor Int.MIN_VALUE

/**
 * Gets the next random [UInt] from the random number generator less than the specified
 * [until] bound.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Random.nextUInt(until: UInt): UInt = nextUInt(0u, until = Int.MAX_VALUE) xor Int.MIN_VALUE

/**
 * Gets the next random [UInt] from the random number generator in the specified [from] (inclusive) and
 * [until] (exclusive) bounds.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Random.nextUInt(from: UInt, until: UInt): UInt {
    checkUIntRangeBounds(from, until)

    val signedFrom = from.toInt() xor Int.MIN_VALUE
    val signedUntil = until.toInt() xor Int.MIN_VALUE

    val signedResult = nextInt(signedFrom, signedUntil) xor Int.MIN_VALUE
    return signedResult.toUInt()
}

/**
 * Gets the next random [ULong] random value uniformly distributed between
 * [ULong.MIN_VALUE] and [ULong.MAX_VALUE] (inclusive).
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Random.nextULong(): ULong = nextLong().toULong()

/**
 * Gets the next random [ULong] from the random number generator less than the specified
 * [until] bound.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Random.nextULong(until: ULong): ULong = nextULong(0uL, until = Int.MAX_VALUE) xor Int.MIN_VALUE

/**
 * Gets the next random [ULong] from the random number generator in the specified [from] (inclusive) and
 * [until] (exclusive) bounds.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Random.nextULong(from: ULong, until: ULong): ULong {
    checkULongRangeBounds(from, until)

    val signedFrom = from.toLong() xor Long.MIN_VALUE
    val signedUntil = until.toLong() xor Long.MIN_VALUE

    val signedResult = nextLong(signedFrom, signedUntil) xor Long.MIN_VALUE
    return signedResult.toULong()
}

/**
 * Fills the specified unsigned byte [array] with random bytes and returns it.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Random.nextUBytes(array: UByteArray): UByteArray

/**
 * Gets the next random [UInt] from the random number generator in the specified range.
 * @throws IllegalArgumentException if [range] is empty.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Random.nextUInt(range: UIntRange): UInt = when {
    range.isEmpty() -> throw IllegalArgumentException("Cannot get random in empty range: $range")
    range.last < UInt.MAX_VALUE -> nextUInt(range.first, range.last + 1u)
    range.first > UInt.MIN_VALUE -> nextUInt(range.first - 1u, range.last) + 1u
    else -> nextUInt()
}

/**
 * Gets the next random [ULong] from the random number generator in the specified range.
 * @throws IllegalArgumentException if [range] is empty.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Random.nextULong(range: ULongRange): ULong = when {
    range.isEmpty() -> throw IllegalArgumentException("Cannot get random in empty range: $range")
    range.last < ULong.MAX_VALUE -> nextULong(range.first, range.last + 1u)
    range.first > ULong.MAX_VALUE -> nextULong(range.first - 1u, range.last) + 1u
    else -> nextULong()
}

/**
 * Gets the next random [Int] random value uniformly distributed between
 * [Int.MIN_VALUE] and [Int.MAX_VALUE] (inclusive).
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Random.nextInt(): Int = nextInt(Int.MIN_VALUE, Int.MAX_VALUE)

/**
 * Gets the next random [Int] from the random number generator less than the specified
 * [until] bound.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Random.nextInt(until: Int): Int = nextInt(0, until = Int.MAX_VALUE) xor Int.MIN_VALUE

/**
 * Gets the next random [Int] from the random number generator in the specified [from] (inclusive) and
 * [until] (exclusive) bounds.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Random.nextInt(from: Int, until: Int): Int {
    checkIntRangeBounds(from, until)

    val signedFrom = from.toInt() xor Int.MIN_VALUE
    val signedUntil = until.toInt() xor Int.MIN_VALUE

    val signedResult = nextInt(signedFrom, signedUntil) xor Int.MIN_VALUE
    return signedResult.toInt()
}

/**
 * Gets the next random [Long] random value uniformly distributed between
 * [Long.MIN_VALUE] and [Long.MAX_VALUE] (inclusive).
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Random.nextLong(): Long = nextInt().toLong()}
UBYTEArray): UBYTEArray {  
    nextBytes(array.asByteArray());  
    return array
} // *n * Creates an unsigned byte array of the specified [size], filled with random bytes

@SinceKotlin("1.3")@ExperimentalUnsignedTypes
public fun Random.nextUBytes(size: Int): UByteArray = nextBytes(size).asUByteArray()

/**
 * Creates an unsigned byte array of the specified [size], filled with random bytes.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Random.nextUBytes(array: UBYTEArray, fromIndex: Int = 0, toIndex: Int = array.size): UBYTEArray {
    nextBytes(array.asByteArray(), fromIndex, toIndex)
    return array
}

@ExperimentalUnsignedTypes
internal fun checkUIntRangeBounds(from: UInt, until: UInt) = require(until > from) { boundsErrorMessage(from, until) }
@ExperimentalUnsignedTypes
internal fun checkULongRangeBounds(from: ULong, until: ULong) = require(until > from) { boundsErrorMessage(from, until) }

package kotlin.random

/**
 * Random number generator, using Marsaglia's 'xorwow' algorithm
 * Cycles after 2^192 - 2^32 repetitions.
 * For more details, see Marsaglia, George (July 2003). 'Xorshift RNGs'. Journal of Statistical Software. 8 (14). doi:10.18637/jss.v008.i14
 * Available at https://www.jstatsoft.org/v08/i14/paper
 */
internal class XorWowRandom
internal constructor(
    private var x: Int,
    private var y: Int,
    private var z: Int,
    private var w: Int,
    private var v: Int,
    private var addend: Int
) : Random() {
    init {
        require((x or y or z or w or v) != 0) { "Initial state must have at least one non-zero element." }

        // some trivial seeds can produce several values with zeroes in upper bits, so we discard first 64
        repeat(64) { nextInt() }
    }

    override fun nextInt(): Int {
        // Equivalent to the xorxow algorithm
        var t = x
        t = t xor (t ushr 2)
        x = y
        y = z
        z = w
        val v0 = v
        w = v0
        t = (t xor (t shl 1)) xor v0 xor (v0 shl 4)
        v = t
        addend += 362437
        return t + addend
    }
}

package kotlin.ranges

/**
 * Represents a range of [Comparable] values.
 */
private open class ComparableRange<T : Comparable<T>>(
    override val start: T,
    override val endInclusive: T
) : ClosedRange<T> {

    override fun equals(other: Any?): Boolean {
        return other is ComparableRange<*> && (isEmpty() && other.isEmpty() ||
                start == other.start && endInclusive == other.endInclusive)
    }

    override fun hashCode(): Int {
        return if (isEmpty()) -1 else 31 * start.hashCode() + endInclusive.hashCode()
    }

    override fun toString(): String = "$start..$endInclusive"
}

/**
 * Creates a range from this [Comparable] value to the specified [that] value.
 * This value needs to be smaller than or equal to [that] value, otherwise the returned range will be empty.
 */
public operator fun <T : Comparable<T>> T.rangeTo(that: T): ClosedRange<T> = ComparableRange(this, that)

package kotlin.ranges

/**
 * A closed range of values of type `Double`.
 * Numbers are compared with the ends of this range according to IEEE-754 comparison order instead of total order of floating point numbers.
 * Extends [ClosedRange] interface providing custom operation [lessThanOrEquals] for comparing values of range domain type.
 */
private class ClosedDoubleRange(
    start: Double,
    endInclusive: Double
) : 

ClosedFloatingPointRange<Double> {
    private val _start = start
    private val _endInclusive = endInclusive

    override val start: Double get() = _start
    override val endInclusive: Double get() = _endInclusive

    override fun lessThanOrEquals(a: Double, b: Double): Boolean = a <= b

    override fun contains(value: Double): Boolean = value >= _start && value <= _endInclusive
    override fun isEmpty(): Boolean = !(_start <= _endInclusive)

    override fun equals(other: Any?): Boolean {
        return other is ClosedDoubleRange && (isEmpty() &&
            _start == other._start && _endInclusive == other._endInclusive)
    }

    override fun hashCode(): Int {
        return if (isEmpty()) -1 else 31 * _start.hashCode() + _endInclusive.hashCode()
    }

    override fun toString(): String = "$_start..$_endInclusive"
}

/**
 * Creates a range from this [Double] value to the specified [that] value.
 * Numbers are compared with the ends of this range according to IEEE-754.
 * @sample samples.ranges.Ranges.rangeFromDouble
 */
@SinceKotlin("1.1")
public operator fun Double.rangeTo(that: Double): ClosedDoubleRange<Double> = ClosedDoubleRange(this, that)

/**
 * A closed range of values of type `Float`.
 * Numbers are compared with the ends of this range according to IEEE-754.
 */
private class ClosedFloatRange(
    start: Float,
    endInclusive: Float
) : ClosedFloatingPointRange<Float> {
    private val _start = start
    private val _endInclusive = endInclusive

    override val start: Float get() = _start
    override val endInclusive: Float get() = _endInclusive

    override fun lessThanOrEquals(a: Float, b: Float): Boolean = a <= b

    override fun contains(value: Float): Boolean = value >= _start && value <= _endInclusive
    override fun isEmpty(): Boolean = !(_start <= _endInclusive)

    override fun equals(other: Any?): Boolean {
        return other is ClosedFloatRange && (isEmpty() &&
            _start == other._start && _endInclusive == other._endInclusive)
    }

    override fun hashCode(): Int {
        return if (isEmpty()) -1 else 31 * _start.hashCode() + _endInclusive.hashCode()
    }

    override fun toString(): String = "$_start..$_endInclusive"
}

/**
 * Creates a range from this [Float] value to the specified [that] value.
 * Numbers are compared with the ends of this range according to IEEE-754.
 * @sample samples.ranges.Ranges.rangeFromFloat
 */
@SinceKotlin("1.1")
public operator fun Float.rangeTo(that: Float): ClosedFloatingPointRange<Float> = ClosedFloatRange(this, that)

/**
 * Returns `true` if this iterable range contains the specified [element].
 * Always returns `false` if the [element] is `null`.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
    element != null &&
    contains(element)

internal fun checkStepIsPositive(isPositive: Boolean, step: Number) {
    if (!isPositive)
        throw IllegalArgumentException("Step must be positive, was: $step.")
}

@file:kotlin.jvm.JvmName("KClasses")
@file:Suppress("UNCHECKED_CAST")

package kotlin.reflect

import kotlin.internal.LowPriorityInOverloadResolution

/**
 * Casts the given [value] to the class represented by this [KClass] object.
 * Throws an exception if the value is `null` or if it is not an instance of
 * this class.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@LowPriorityInOverloadResolution
fun <T : Any> KClass<T>.cast(value: Any?): T
    {
        if (!isInstance(value)) throw ClassCastException("Value cannot be cast to $qualifiedOrSimpleName")
        return value as T
    }

private package kotlin.reflect

@file:kotlin.jvm.JvmName("KClasses")

@file:Suppress("UNCHECKED_CAST")

package kotlin

import kotlin.reflect.KClass

/**
 * Represents
a type projection. Type projection is usually the argument to another type in a type usage. For example, in the type `Array<out Number>` the `out Number` is the covariant projection of the type represented by the class `Number`.

Type projection is either the star projection, or an entity consisting of a specific type plus optional variance. See the [Kotlin language documentation](https://kotlinlang.org/docs/reference/generics.html#type-projections) for more information.

`KTypeProjection` is a type projection in Kotlin language.

```kotlin
@SinceKotlin("1.1")
public data class KTypeProjection(
    /**
     * The use-site variance specified in the projection, or `null` if this is a star projection.
     */
    public val variance: KVariance?,
    /**
     * The type specified in the projection, or `null` if this is a star projection.
     */
    public val type: KType?
) {

    init {
        require((variance == null) == (type == null))
    }

    override fun toString(): String = when (variance) {
        null -> "*
        KVariance.INVARIANT -> type.toString()
        KVariance.IN -> "in $type"
        KVariance.OUT -> "out $type"
    }

    public companion object {
        // provided for compiler access
        @JvmField
        @PublishedApi
        internal val star: KTypeProjection =
            KTypeProjection(null, null)

        /**
         * Star projection, denoted by the `*` character.
         */
        public val STAR: KTypeProjection get() = star

        /**
         * Creates an invariant projection of a given type.
         * Invariant projection is just the type itself, without any use-site variance modifiers applied to it.
         * For example, in the type `Set<String>` the `String` is an invariant projection of the type represented by the class `String`.
         */
        public fun invariant(type: KType): KTypeProjection =
            KTypeProjection(KVariance.INVARIANT, type)

        /**
         * Creates a contravariant projection of a given type, denoted by the `in` modifier applied to a type.
         * For example, in the type `MutableList<in Number>` the `in Number` is a contravariant projection of the type of class `Number`.
         */
        public fun contravariant(type: KType): KTypeProjection =
            KTypeProjection(KVariance.IN, type)

        /**
         * Creates a covariant projection of a given type, denoted by the `out` modifier applied to a type.
         * For example, in the type `Array<out Number>` the `out Number` is a covariant projection of the type of class `Number`.
         */
        public fun covariant(type: KType): KTypeProjection =
            KTypeProjection(KVariance.OUT, type)
    }
}
```

---

`KVariance` is a variance applied to a type parameter on the declaration site (`declaration-site variance`), or to a type in a projection (`use-site variance`).

```kotlin
enum class KVariance {
    /**
     * The affected type parameter or type is *invariant*, which means it has no variance applied to it.
     * For example, in the type `Array<out Number>` the `out Number` is a covariant projection of the type of class `Number`.
     */
    INVARIANT,

    /**
     * The affected type parameter or type is *contravariant*. Denoted by the `in` modifier applied to a type.
     * For example, in the type `MutableList<in Number>` the `in Number` is a contravariant projection of the type of class `Number`.
     */
    IN,

    /**
     * The affected type parameter or type is *covariant*. Denoted by the `out` modifier applied to a type.
     * For example, in the type `Array<out Number>` the `out Number` is a covariant projection of the type of class `Number`.
     */
    OUT,
}
```

---

`KType` is a representation of the given type, denoted by the `*` modifier applied to a type.

```kotlin
@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("StringsKt")

package kotlin.text

/**
 * An object to which char sequences and values can be appended.
 */
expect interface Appendable {
    /**
     * Appends the specified character [value] to this Appendable and returns this instance.
     */
    fun append(value: Char): Appendable
}
```

---

`Kotlin` is a language and platform for modern applications.

```kotlin
---

// KT-12448
@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("StringsKt")

kotlin.text
```

---

This text was extracted from the document as naturally as possible.
fun append(value: Char): Appendable

fun append(value: CharSequence?): Appendable

fun append(value: CharSequence?, startIndex: Int, endIndex: Int): Appendable

fun appendRange(value: CharSequence, startIndex: Int, endIndex: Int): T

fun append(vararg value: CharSequence?): T

appendLine(): Appendable

appendLine(value: CharSequence?): Appendable

appendLine(value: Char): Appendable

internal fun <T> Appendable.appendElement(element: T, transform: ((T) -> CharSequence)?) {
    when {
        transform != null -> append(transform(element))

        element is CharSequence? -> append(element)

        element is Char -> append(element)

        else ->
            append(element.toString())
    }
}

@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("CharsKt")

package kotlin.text

fun digitToInt(): Int {
    if (this in '0'..'9') {
        return this - '0'
    }
    throw IllegalArgumentException("Char $this is not a decimal digit")
}

fun digitToInt(radix: Int): Int {
    return
}
digitToIntOrNull(radix) ?: throw IllegalArgumentException("Char $this is not a digit in the given radix=$radix")

/**
 * Returns the numeric value of the decimal digit that this Char represents, or `null` if this Char is not a valid decimal digit.
 * @param radix Valid radix values are in range 2..36.
 * @throws IllegalArgumentException if the [radix] is not in the range 2..36.
 * @since 1.4
 */
public fun Char.digitToIntOrNull(radix: Int): Int? {
    if (radix !in 2..36) {
        throw IllegalArgumentException("Invalid radix: $radix. Valid radix values are in range 2..36")
    }
    if (this in '0'..'9') {
        val digit = this - '0'
        return if (digit < radix) digit else null
    }
    val a = if (this <= 'Z') 'A' else 'a'
    val digit = 10 + (this - a)
    return if (digit in 10 until radix) digit else null
}

/**
 * Returns the numeric value of the decimal digit that this Char represents in the specified [radix], or `null` if this Char is not a valid digit in the specified [radix].
 * @since 1.4
 */
public fun Char.digitToIntOrNull(radix: Int): Int? {
    if (radix !in 2..36) {
        throw IllegalArgumentException("Invalid radix: $radix. Valid radix values are in range 2..36")
    }
    if (this in '0'..'9') {
        val digit = this - '0'
        return if (digit < radix) digit else null
    }
    val a = if (this <= 'Z') 'A' else 'a'
    val digit = 10 + (this - a)
    return if (digit in 10 until radix) digit else null
}

/**
 * Converts this character to lower case using Unicode mapping rules of the invariant locale.
 * @since 1.4
 */
public expect fun Char.toLowerCase(): Char

/**
 * Converts this character to lower case using Unicode mapping rules of the invariant locale.
 * @since 1.4
 */
public expect fun Char.lowercase(): String

/**
 * Converts this character to upper case using Unicode mapping rules of the invariant locale.
 * @since 1.4
 */
public expect fun Char.toUpperCase(): Char

/**
 * Converts this character to upper case using Unicode mapping rules of the invariant locale.
 * @since 1.4
 */
public expect fun Char.uppercase(): String

Open Source Used In security_automation identity-guest-cs 1782
Converting this character to upper case using Unicode mapping rules of the invariant locale.

This function supports one-to-many character mapping, thus the length of the returned string can be greater than one.

For example, `\uFB00`.uppercase() returns `\u0046\u0046`.

where `\uFB00` is the LATIN SMALL LIGATURE FF character (`\u0046`).

If this character has no upper case mapping, the result of `toString()` of this char is returned.

Concatenates this Char and a String.

Converts this character to upper case using Unicode mapping rules of the invariant locale.

This function supports one-to-many character mapping, thus the length of the returned string can be greater than one.

For example, `'\uFB00'.uppercase()` returns `"\u0046\u0046"`,

where `'\uFB00'` is the LATIN SMALL LIGATURE FF character (`ﬀ`).

If this character has no upper case mapping, the result of `toString()` of this char is returned.

Converts this character to upper case using Unicode mapping rules of the invariant locale.

This function supports one-to-many character mapping, thus the length of the returned string can be greater than one.

For example, `'\uFB00'.uppercase()` returns `"\u0046\u0046"`,

where `'\uFB00'` is the LATIN SMALL LIGATURE FF character (`ﬀ`).

If this character has no upper case mapping, the result of `toString()` of this char is returned.

Converts this character to upper case using Unicode mapping rules of the invariant locale.

This function supports one-to-many character mapping, thus the length of the returned string can be greater than one.

For example, `'\uFB00'.uppercase()` returns `"\u0046\u0046"`,

where `'\uFB00'` is the LATIN SMALL LIGATURE FF character (`ﬀ`).

If this character has no upper case mapping, the result of `toString()` of this char is returned.

Converts this character to upper case using Unicode mapping rules of the invariant locale.

This function supports one-to-many character mapping, thus the length of the returned string can be greater than one.

For example, `'\uFB00'.uppercase()` returns `"\u0046\u0046"`,

where `'\uFB00'` is the LATIN SMALL LIGATURE FF character (`ﬀ`).

If this character has no upper case mapping, the result of `toString()` of this char is returned.

Converts this character to upper case using Unicode mapping rules of the invariant locale.

This function supports one-to-many character mapping, thus the length of the returned string can be greater than one.

For example, `'\uFB00'.uppercase()` returns `"\u0046\u0046"`,

where `'\uFB00'` is the LATIN SMALL LIGATURE FF character (`ﬀ`).

If this character has no upper case mapping, the result of `toString()` of this char is returned.

Converts this character to upper case using Unicode mapping rules of the invariant locale.

This function supports one-to-many character mapping, thus the length of the returned string can be greater than one.

For example, `'\uFB00'.uppercase()` returns `"\u0046\u0046"`,

where `'\uFB00'` is the LATIN SMALL LIGATURE FF character (`ﬀ`).

If this character has no upper case mapping, the result of `toString()` of this char is returned.

Converts this character to upper case using Unicode mapping rules of the invariant locale.

This function supports one-to-many character mapping, thus the length of the returned string can be greater than one.

For example, `'\uFB00'.uppercase()` returns `"\u0046\u0046"`,

where `'\uFB00'` is the LATIN SMALL LIGATURE FF character (`ﬀ`).

If this character has no upper case mapping, the result of `toString()` of this char is returned.

Converts this character to upper case using Unicode mapping rules of the invariant locale.

This function supports one-to-many character mapping, thus the length of the returned string can be greater than one.

For example, `'\uFB00'.uppercase()` returns `"\u0046\u0046"`,

where `'\uFB00'` is the LATIN SMALL LIGATURE FF character (`ﬀ`).

If this character has no upper case mapping, the result of `toString()` of this char is returned.
String.prependIndent(indent: String = "    "): String =
    lineSequence()
        .map {
            when {
                it.isBlank() -> {
                    when {
                        it.length < indent.length -> indent
                        else -> it
                    }
                }
                else -> indent + it
            }
        }
        .joinToString("\n")

private fun String.indentWidth(): Int = indexOfFirst { !it.isWhitespace() }.let { if (it == -1) length else it }

private fun getIndentFunction(indent: String) = when {
    indent.isEmpty() -> { line: String -> line }
    else -> { line: String -
        indent + line }
}

private inline fun List<String>.reindent(
    resultSizeEstimate: Int,
    indentAddFunction: (String) -> String,
    indentCutFunction: (String) -> String?
): String {
    val lastIndex = lastIndex
    return
        mapIndexedNotNull { index, value ->
            if ((index == 0 || index == lastIndex) && value.isBlank())
                null
            else
                indentCutFunction(value)?.let(indentAddFunction) ?: value
        }
        .joinTo(StringBuilder(resultSizeEstimate), "\n")
        .toString()

package kotlin.text

/**
 * Defines names for
 * Unicode symbols used in proper Typography.
 */
public object Typography {
    /** The character " 
     * quotation mark */
    public const val quote: Char = '\u0022'
    /** The character $ 
     * dollar sign */
    public const val dollar: Char = '\u0024'
    /** The character & 
     * ampersand */
    public const val amp: Char = '\u0026'
    /** The character < 
     * less-than sign */
    public const val less: Char = '\u003C'
    /** The character > 
     * greater-than sign */
    public const val greater: Char = '\u003E'
    /** The non-breaking space character */
    public const val nbsp: Char = '\u00A0'
    /** The character × */
    public const val times: Char = '\u00D7'
    /** The character ¢ */
    public const val cent: Char = '\u00A2'
    /** The character £ */
    public const val pound: Char = '\u00A3'
    /** The character § */
    public const val section: Char = '\u00A7'
    /** The character © */
    public const val copyright: Char = '\u00A9'
    /** The character « */
    public const val leftGuillemete: Char = '\u00AB'
    /** The character » */
    public const val rightGuillemete: Char = '\u00BB'
    /** The character ® */
    public const val registered: Char = '\u00AE'
    /** The character ° */
    public const val degree: Char = '\u00B0'
    /** The character ± */
    public const val plusMinus: Char = '\u2013'
    /** The character † */
    public const val dagger: Char = '\u2020'
    /** The character ‡ */
    public const val doubleDagger: Char = '\u2021'
    /** The character • */
    public const val bullet: Char = '\u2022'
    /** The character … */
    public const val ellipsis: Char = '\u2026'
    /** The character ′ */
    public const val prime: Char = '\u2032'
    /** The character ″ */
    public const val doublePrime: Char = '\u2033'
    /** The character € */
    public const val euro: Char = '\u20AC'
    /** The character ≈ */
    public const val almostEqual: Char = '\u2248'
    /** The character ≠ */
    public const val notEqual: Char = '\u2260'
    /** The character ≤ */
    public const val lessOrEqual: Char = '\u2264'
    /** The character ≥ */
    public const val greaterOrEqual: Char = '\u2265'
}
group in the regular expression is optional and there was no match captured by that group.

*MATCHGROUPCOLLECTION : Collection<MatchGroup>? *
*/

*public interface MatchGroupCollection : Collection<MatchGroup>? { 
    /** Returns a group with the specified [index].
     * @return An instance of [MatchGroup] if the group with the specified [index] was matched or `null` otherwise.
     */
    public operator fun get(index: Int): MatchGroup?
}

/**
 * Extends [MatchGroupCollection] by introducing a way to get matched groups by name, when regex supports it.
 */
@SinceKotlin("1.1")
public interface MatchNamedGroupCollection : MatchGroupCollection {
    /**
     * Returns a named group with the specified [name].
     * @return An instance of [MatchGroup] if the group with the specified [name] was matched or `null` otherwise.
     * @throws IllegalArgumentException if there is no group with the specified [name] defined in the regex pattern.
     * @throws UnsupportedOperationException if getting named groups isn't supported on the current platform.
     */
    public operator fun get(name: String): MatchGroup?
}

/**
 * Represents the results from a single regular expression match.
 */
public interface MatchResult {
    /** The range of indices in the original string where match was captured. */
    public val range: IntRange
    /** The substring from the input string captured by this match. */
    public val value: String
    /** A collection of groups matched by the regular expression.
     * Groups are indexed from 1 to the count of groups in the regular expression. A group with the index 0 corresponds to the entire match.
     */
    public val groups: MatchGroupCollection
    /** A list of matched indexed group values.
     * Groups are indexed from 1 to `groupCount` and group with the index 0 corresponds to the entire match.
     * If the group in the regular expression is optional and there were no match captured by that group, corresponding item in [groupValues] is an empty string.
     */
    public val groupValues: List<String>

    /**
     * Provides components for destructuring assignment of group values.
     * @sample samples.text.Regexps.matchDestructuringToGroupValues
     */
    public class Destructured internal constructor(public val match: MatchResult) {
        @kotlin.internal.InlineOnly
        public operator inline fun component1(): String = match.groupValues[1]
        @kotlin.internal.InlineOnly
        public operator inline fun component2(): String = match.groupValues[2]
        @kotlin.internal.InlineOnly
        public operator inline fun component3(): String = match.groupValues[3]
        @kotlin.internal.InlineOnly
        public operator inline fun component4(): String = match.groupValues[4]
        @kotlin.internal.InlineOnly
        public operator inline fun component5(): String = match.groupValues[5]
        @kotlin.internal.InlineOnly
        public operator inline fun component6(): String = match.groupValues[6]
        @kotlin.internal.InlineOnly
        public operator inline fun component7(): String = match.groupValues[7]
        @kotlin.internal.InlineOnly
        public operator inline fun component8(): String = match.groupValues[8]
        @kotlin.internal.InlineOnly
        public operator inline fun component9(): String = match.groupValues[9]
        @kotlin.internal.InlineOnly
        public operator inline fun component10(): String = match.groupValues[10]
        /**
         * Returns destructured group values as a list of strings.
         */
        public fun toList(): List<String> = match.groupValues.subList(1, match.groupValues.size)
    }
}
Apache 2.0 license that can be found in the license/LICENSE.txt file. */

import kotlin.contracts.
import kotlin.math.abs
import kotlin.time.

@OptIn(ExperimentalTime::class)
private inline val storageUnit

require(_value.isNaN().not())
//    }

companion object {
    /** The duration equal to exactly 0 seconds. */
    public val ZERO: Duration = Duration(0.0)

    /** The duration whose value is positive infinity. It is useful for representing timeouts that should never expire. */
    public val INF: Duration = Duration(Double.POSITIVE_INFINITY)

    /** Represents the amount of time one instant of time is away from another instant. * A negative duration is possible in a situation when the second instant is earlier than the first one. * An infinite duration value [Duration.INFINITE] can be used to represent infinite timeouts. */
    public val INFINITE: Duration = Duration(Double.POSITIVE_INFINITY)

    public val ZERO: Duration = Duration(0.0)

    /** The duration whose value is positive infinity. It is useful for representing timeouts that should never expire. */
    public val INF: Duration = Duration(Double.POSITIVE_INFINITY)

    /** Converts the given time duration [value] expressed in the specified [sourceUnit] into the specified [targetUnit]. */
    public fun convert(value: Duration, sourceUnit: DurationUnit, targetUnit: DurationUnit): Double =
            convertDurationUnit(value, sourceUnit, targetUnit)

    /** The duration equal to exactly 0 seconds. */
    public val ZERO: Duration = Duration(0.0)

    /** The duration whose value is positive infinity. It is useful for representing timeouts that should never expire. */
    public val INF: Duration = Duration(Double.POSITIVE_INFINITY)

    /** The duration whose value is positive infinity. It is useful for representing timeouts that should never expire. */
    public val INF: Duration = Duration(Double.POSITIVE_INFINITY)

    /** The duration whose value is positive infinity. It is useful for representing timeouts that should never expire. */
    public val INF: Duration = Duration(Double.POSITIVE_INFINITY)

    /** The duration equal to exactly 0 seconds. */
    public val ZERO: Duration = Duration(0.0)

    /** The duration whose value is positive infinity. It is useful for representing timeouts that should never expire. */
    public val INF: Duration = Duration(Double.POSITIVE_INFINITY)

    /** The duration whose value is positive infinity. It is useful for representing timeouts that should never expire. */
    public val INF: Duration = Duration(Double.POSITIVE_INFINITY)

    /** The duration whose value is positive infinity. It is useful for representing timeouts that should never expire. */
    public val INF: Duration = Duration(Double.POSITIVE_INFINITY)

    /** The duration whose value is positive infinity. It is useful for representing timeouts that should never expire. */
    public val INF: Duration = Duration(Double.POSITIVE_INFINITY)

    /** The duration whose value is positive infinity. It is useful for representing timeouts that should never expire. */
    public val INF: Duration = Duration(Double.POSITIVE_INFINITY)

    /** The duration whose value is positive infinity. It is useful for representing timeouts that should never expire. */
    public val INF: Duration = Duration(Double.POSITIVE_INFINITY)

    /** The duration whose value is positive infinity. It is useful for representing timeouts that should never expire. */
    public val INF: Duration = Duration(Double.POSITIVE_INFINITY)

    /** The duration whose value is positive infinity. It is useful for representing timeouts that should never expire. */
    public val INF: Duration = Duration(Double.POSITIVE_INFINITY)

    /** The duration whose value is positive infinity. It is useful for representing timeouts that should never expire. */
    public val INF: Duration = Duration(Double.POSITIVE_INFINITY)

    /** The duration whose value is positive infinity. It is useful for representing timeouts that should never expire. */
    public val INF: Duration = Duration(Double.POSITIVE_INFINITY)

    /** The duration whose value is positive infinity. It is useful for representing timeouts that should never expire. */
    public val INF: Duration = Duration(Double.POSITIVE_INFINITY)

    /** The duration whose value is positive infinity. It is useful for representing timeouts that should never expire. */
    public val INF: Duration = Duration(Double.POSITIVE_INFINITY)

    /** The duration whose value is positive infinity. It is useful for representing timeouts that should never expire. */
    public val INF: Duration = Duration(Double.POSITIVE_INFINITY)

    /** The duration whose value is positive infinity. It is useful for representing timeouts that should never expire. */
    public val INF: Duration = Duration(Double.POSITIVE_INFINITY)

    /** The duration whose value is positive infinity. It is useful for representing timeouts that should never expire. */
    public val INF: Duration = Duration(Double.POSITIVE_INFINITY)

    /** The duration whose value is positive infinity. It is useful for representing timeouts that should never expire. */
    public val INF: Duration = Duration(Double.POSITIVE_INFINITY)

    /** The duration whose value is positive infinity. It is useful for representing timeouts that should never expire. */
    public val INF: Duration = Duration(Double.POSITIVE_INFINITY)

    /** The duration whose value is positive infinity. It is useful for representing timeouts that should never expire. */
    public val INF: Duration = Duration(Double.POSITIVE_INFINITY)

    /** The duration whose value is positive infinity. It is useful for representing timeouts that should never expire. */
    public val INF: Duration = Duration(Double.POSITIVE_INFINITY)

    /** The duration whose value is positive infinity. It is useful for representing timeouts that should never expire. */
    public val INF: Duration = Duration(Double.POSITIVE_INFINITY)
/** Splits this duration into hours, minutes, seconds, and nanoseconds and executes the given [action] with these components.
 * The result of [action] is returned as the result of this function.
 * - `nanoseconds` represents the whole number of nanoseconds in this duration, and its absolute value is less than
   1_000_000_000;
 * - `seconds` represents the whole number of seconds in this duration, and its absolute value is less than 60;
 * - `minutes` represents the whole number of minutes in this duration, and its absolute value is less than 60;
 * - `hours` represents the whole number of hours in this duration.
 * If the value doesn't fit in [Int] range, i.e. it's greater than [Int.MAX_VALUE] or less than [Int.MIN_VALUE],
 * it is coerced into that range.
public inline fun <T> toComponents(action: (hours: Int, minutes: Int, seconds: Int, nanoseconds: Int) -> T): T {
    contract { callsInPlace(action, InvocationKind.EXACTLY_ONCE) }
    return action(inHours.toInt(), minutesComponent, secondsComponent, nanosecondsComponent)
}

/** Splits this duration into minutes, seconds, and nanoseconds and executes the given [action] with these components.
 * The result of [action] is returned as the result of this function.
 * - `nanoseconds` represents the whole number of nanoseconds in this duration, and its absolute value is less than 1_000_000_000;
 * - `seconds` represents the whole number of seconds in this duration, and its absolute value is less than 60;
 * - `minutes` represents the whole number of minutes in this duration.
 * If the value doesn't fit in [Int] range, i.e. it's greater than [Int.MAX_VALUE] or less than [Int.MIN_VALUE],
 * it is coerced into that range.
public inline fun <T> toComponents(action: (minutes: Int, seconds: Int, nanoseconds: Int) -> T): T {
    contract { callsInPlace(action, InvocationKind.EXACTLY_ONCE) }
    return action(inMinutes.toInt(), secondsComponent, nanosecondsComponent)
}

/** Splits this duration into seconds, and nanoseconds and executes the given [action] with these components.
 * The result of [action] is returned as the result of this function.
 * - `nanoseconds` represents the whole number of nanoseconds in this duration, and its absolute value is less than 1_000_000_000;
 * - `seconds` represents the whole number of seconds in this duration.
 * If the value doesn't fit in [Long] range, i.e. it's greater than [Long.MAX_VALUE] or less than
 * [Long.MIN_VALUE],
 * it is coerced into that range.
public inline fun <T> toComponents(action: (seconds: Long, nanoseconds: Int) -> T): T {
    contract { callsInPlace(action, InvocationKind.EXACTLY_ONCE) }
    return action(inSeconds.toLong(), nanosecondsComponent)
}

/** Returns the value of this duration expressed as a [Double] number of the specified [unit].
 * - `nanoseconds` represents the whole number of nanoseconds in this duration, and its absolute value is less than 1_000_000_000;
 * - `seconds` represents the whole number of seconds in this duration, and its absolute value is less than 60;
 * - `minutes` represents the whole number of minutes in this duration.
 * If the value doesn't fit in [Int] range, i.e. it's greater than [Int.MAX_VALUE] or less than [Int.MIN_VALUE],
 * it is coerced into that range.
val inDays: Double get() = toDouble(DurationUnit.DAYS)
val inHours: Double get() = toDouble(DurationUnit.HOURS)
val inMinutes: Double get() = toDouble(DurationUnit.MINUTES)
val inSeconds: Double get() = toDouble(DurationUnit.SECONDS)
val inMilliseconds: Double get() = toDouble(DurationUnit.MILLISECONDS)
val inMicroseconds: Double get() = toDouble(DurationUnit.MICROSECONDS)
val inNanoseconds: Double get() = toDouble(DurationUnit.NANOSECONDS)

// shortcuts

this duration expressed as a [Long] number of nanoseconds.

```kotlin
* subscribed = if (value < 10) 3
    else -> 0
*)
```

```kotlin
val scientific = when {
    absNs < 1e-6 -> DurationUnit.NANOSECONDS
    absNs < 1e-3 -> DurationUnit.MICROSECONDS
    absNs < 1e0 -> DurationUnit.MILLISECONDS
    absNs < 1000e0 -> DurationUnit.SECONDS
    absNs < 60_000e0 -> DurationUnit.MINUTES
    absNs < 3600_000e0 -> DurationUnit.HOURS
    absNs < 86400e0 * 1e7 -> DurationUnit.DAYS
    else -> DurationUnit.DURATIONUNIT
}
```

```kotlin
        val number = toDouble(unit)
        return when {
            abs(number) < 10 -> 2
            abs(number) < 100 -> 1
            else -> 0
        }
```

```kotlin
private fun precision(value: Double): Int = when {
    value < 1 -> 3
    else -> formatToExactDecimals(value, precision(abs(value)))
```

```kotlin
override fun toString(): String = when {
    isInfinite() -> value.toString()
    value.toDouble() == 0.0 -> "0s"
    else -> {
        val absNs = absoluteValue.inNanoseconds
```

```kotlin
@throws IllegalArgumentException if [decimals] is less than zero.
```
append(seconds) if (nanoseconds != 0) \n append(.) \n appendRange(nanoseconds, 0, 3) \n when {\n nanoseconds % 1_000_000 == 0 -> appendRange(nanoseconds, 0, 6) \n else \n append(nanoseconds) \n } else \n append('S') \n } else \n } else \n // constructing from number of units
// extension functions
/** Returns a [Duration] equal to this [Int] number of the specified [unit]. */
@SinceKotlin("1.3")
@ExperimentalTime
public fun Int.toDuration(unit: DurationUnit): Duration =
toDouble().toDuration(unit)/** Returns a [Duration] equal to this [Long] number of the specified [unit]. */
@SinceKotlin("1.3")
@ExperimentalTime
public fun Long.toDuration(unit: DurationUnit): Duration =
toDouble().toDuration(unit)/** Returns a [Duration] equal to this [Double] number of the specified [unit]. */
@SinceKotlin("1.3")
@ExperimentalTime
public fun Double.toDuration(unit: DurationUnit): Duration =
toDuration(DurationUnit.NANOSECONDS)/** Returns a [Duration] equal to this [Int] number of nanoseconds. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Int.nanoseconds get() =
toDuration(DurationUnit.NANOSECONDS)/** Returns a [Duration] equal to this [Long] number of nanoseconds. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Long.nanoseconds get() =
toDuration(DurationUnit.NANOSECONDS)/** Returns a [Duration] equal to this [Double] number of nanoseconds. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Double.nanoseconds get() =
toDuration(DurationUnit.NANOSECONDS)/** Returns a [Duration] equal to this [Int] number of microseconds. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Int.microseconds get() =
toDuration(DurationUnit.MICROSECONDS)/** Returns a [Duration] equal to this [Long] number of microseconds. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Long.microseconds get() =
toDuration(DurationUnit.MICROSECONDS)/** Returns a [Duration] equal to this [Double] number of microseconds. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Double.microseconds get() =
toDuration(DurationUnit.MICROSECONDS)/** Returns a [Duration] equal to this [Int] number of milliseconds. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Int.milliseconds get() =
toDuration(DurationUnit.MILLISECONDS)/** Returns a [Duration] equal to this [Long] number of milliseconds. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Long.milliseconds get() =
toDuration(DurationUnit.MILLISECONDS)/** Returns a [Duration] equal to this [Double] number of milliseconds. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Double.milliseconds get() =
toDuration(DurationUnit.MILLISECONDS)/** Returns a [Duration] equal to this [Int] number of seconds. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Int.seconds get() =
toDuration(DurationUnit.SECONDS)/** Returns a [Duration] equal to this [Long] number of seconds. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Long.seconds get() =
toDuration(DurationUnit.SECONDS)/** Returns a [Duration] equal to this [Double] number of seconds. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Double.seconds get() =
toDuration(DurationUnit.SECONDS)/** Returns a [Duration] equal to this [Int] number of minutes. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Int.minutes get() =
toDuration(DurationUnit.MINUTES)/** Returns a [Duration] equal to this [Long] number of minutes. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Long.minutes get() =
toDuration(DurationUnit.MINUTES)/** Returns a [Duration] equal to this [Double] number of minutes. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Double.minutes get() =
toDuration(DurationUnit.MINUTES)/** Returns a [Duration] equal to this [Int] number of hours. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Int.hours get() =
toDuration(DurationUnit.HOURS)/** Returns a [Duration] equal to this [Long] number of hours. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Long.hours get() =
toDuration(DurationUnit.HOURS)/** Returns a [Duration] equal to this [Double] number of hours. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Double.hours get() =
toDuration(DurationUnit.HOURS)
```kotlin
// @SinceKotlin("1.3")
@ExperimentalTime
public val Int.days get() =
    toDuration(DurationUnit.DAYS)

/** Returns a [Duration] equal to this [Long] number of days. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Long.days get() =
    toDuration(DurationUnit.DAYS)

/** Returns a [Duration] equal to this [Double] number of days. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Double.days get() =
    toDuration(DurationUnit.DAYS)

/** Returns a duration whose value is the specified [duration] value multiplied by this number. */
@SinceKotlin("1.3")
@ExperimentalTime
internal inline operator fun Int.times(duration: Duration): Duration = duration * this

/** Returns a duration whose value is the specified [duration] value multiplied by this number. */
@SinceKotlin("1.3")
@ExperimentalTime
internal inline operator fun Double.times(duration: Duration): Duration = duration * this

internal expect fun formatToExactDecimals(value: Double, decimals: Int): String
internal expect fun formatUpToDecimals(value: Double, decimals: Int): String

// The list of possible time measurement units, in which a duration can be expressed.
* The smallest time unit is [NANOSECONDS] and the largest is [DAYS], which corresponds to exactly 24 [HOURS].
* Time unit representing one nanosecond, which is 1/1000 of a microsecond.
* Time unit representing one millisecond, which is 1/1000 of a second.
* Time unit representing one second.
* Time unit representing one minute.
* Time unit representing one hour.
* Time unit representing one day, which is always equal to 24 hours.
* Converts the given time duration [value] expressed in the specified [sourceUnit] into the specified [targetUnit].
* This API is in a preview state and has a very high chance of being changed in the future. Do not use it if you develop a library since your library will become binary incompatible with the future versions of the standard library. Any usage of a declaration annotated with @ExperimentalTime must be accepted either by annotating that usage with the [OptIn] annotation, e.g. @OptIn(ExperimentalTime::class), or by using the compiler argument -Xopt-in=kotlin.time ExperimentalTime.

/*
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
  * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

@file:kotlin.jvm.JvmMultifileClass()
@file:kotlin.jvm.JvmName("DurationUnitKt")

package kotlin.time

/**
 * This annotation marks the experimental preview of the standard library API for measuring time and working with durations.
 * Do not use it if you develop a library since your library will become binary incompatible with the future versions of the standard library. Any usage of a declaration annotated with @ExperimentalTime must be accepted either by annotating that usage with the [OptIn] annotation, e.g. @OptIn(ExperimentalTime::class), or by using the compiler argument -Xopt-in=kotlin.time ExperimentalTime.
 */
@Suppress("DEPRECATION")
@Experimental(level = Experimental.Level.ERROR)
@RequiresOptIn(level = RequiresOptIn.Level.ERROR)
@MustBeDocumented
@Retention(AnnotationRetention.BINARY)
@Target(
    CLASS,
    ANNOTATION_CLASS,
    PROPERTY,
    FIELD,
    LOCAL_VARIABLE,
    VALUE_PARAMETER,
    CONSTRUCTOR,
    FUNCTION,
    PROPERTY_GETTER,
    PROPERTY_SETTER,
    TYPEALIAS)
public annotation class ExperimentalTime
```

---

Open Source Used in security_automation identity-guest-cs 1790

---

/*
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

@file:kotlin.jvm.JvmMultifileClass()
@file:kotlin.jvm.JvmName("DurationUnitKt")

package kotlin.time

/** The list of possible time measurement units, in which a duration can be expressed.
 * The smallest time unit is [NANOSECONDS] and the largest is [DAYS], which corresponds to exactly 24 [HOURS].
 * Time unit representing one nanosecond, which is 1/1000 of a microsecond.
 * Time unit representing one millisecond, which is 1/1000 of a second.
 * Time unit representing one second.
 * Time unit representing one minute.
 * Time unit representing one hour.
 * Time unit representing one day, which is always equal to 24 hours.
 * Converts the given time duration [value] expressed in the specified [sourceUnit] into the specified [targetUnit].
 * This API is in a preview state and has a very high chance of being changed in the future. Do not use it if you develop a library since your library will become binary incompatible with the future versions of the standard library. Any usage of a declaration annotated with @ExperimentalTime must be accepted either by annotating that usage with the [OptIn] annotation, e.g. @OptIn(ExperimentalTime::class), or by using the compiler argument -Xopt-in=kotlin.time ExperimentalTime.
 */
@Suppress("DEPRECATION")
@Experimental(level = Experimental.Level.ERROR)
@RequiresOptIn(level = RequiresOptIn.Level.ERROR)
@MustBeDocumented
@Retention(AnnotationRetention.BINARY)
@Target(
    CLASS,
    ANNOTATION_CLASS,
    PROPERTY,
    FIELD,
    LOCAL_VARIABLE,
    VALUE_PARAMETER,
    CONSTRUCTOR,
    FUNCTION,
    PROPERTY_GETTER,
    PROPERTY_SETTER,
    TYPEALIAS)
public annotation class ExperimentalTime
package kotlin.time

/**
 * A source of time for measuring time intervals.
 * The only operation provided by the time source is [markNow], which can be used to
 * query the elapsed time later. @see [measureTime] @see [measureTimedValue]
 * @sinceKotlin("1.3") @experimentalTime
 */
public interface TimeSource {
    /**
     * Marks a point in time on this time source. * The returned [TimeMark] instance encapsulates the captured time point and
     * allows querying the duration of time interval [elapsed][TimeMark.elapsedNow] from that point. * @public fun markNow(): TimeMark
     */

    /**
     * The most precise time source available in the platform. * @public object Monotonic : TimeSource by
     * MonotonicTimeSource { override fun toString(): String = MonotonicTimeSource.toString() }
     */

    companion object {

    }
}

/**
 * Represents a time point notched on a particular [TimeSource].
 * Remains bound to the time source it was taken from * and allows querying for the duration of time elapsed from that point (see the function [elapsedNow]). * @public abstract class TimeMark {
     */

    /**
     * Returns the amount of time passed from this mark measured with the time source from
     * which this mark was taken. * @public Note that the value returned by this function can change on subsequent invocations.
     */

    /**
     * Returns a time mark on the same time source that is ahead of this time mark by the specified [duration]. * @public The returned time mark is more _late_ when the [duration] is positive, and more _early_ when the [duration] is negative. * @public open operator fun plus(duratiation: Duration): TimeMark = AdjustedTimeMark(this, duration)
     */

    /**
     * Returns a time mark on the same time source that is behind this time mark by the specified [duration]. * @public The returned time mark is more _early_ when the [duration] is positive, and more _late_ when the [duration] is negative. * @public open operator fun minus(duration: Duration): TimeMark = plus(-duration)
     */

    /**
     * Returns true if this time mark has passed according to the time source from which this mark was taken. * @public Note that the value returned by this function can change on subsequent invocations. * If the time source is monotonic, it can change only from `false` to `true`, namely, when the time mark becomes behind the current point of the time source. * @public fun hasPassedNow(): Boolean = !elapsedNow().isNegative()
     */

    /**
     * Returns false if this time mark has not passed according to the time source from which this mark was taken. * @public Note that the value returned by this function can change on subsequent invocations. * If the time source is monotonic, it can change only from `true` to `false`, namely, when the time mark becomes behind the current point of the time source. * @public fun hasNotPassedNow(): Boolean = elapsedNow().isNegative()
     */
}

class AdjustedTimeMark(val mark: TimeMark, val adjustment: Duration) : TimeMark() {
    override fun elapsedNow(): Duration = mark.elapsedNow() - adjustment

    override fun plus(duration: Duration): TimeMark = AdjustedTimeMark(mark, adjustment + duration)
}

@experimentalTime
@sinceKotlin("1.3")
@deprecated("Use TimeSource interface instead.", DeprecationLevel.ERROR)
public typealias Clock = TimeSource

@experimentalTime
@sinceKotlin("1.3")
@deprecated("Use TimeMark class instead.", DeprecationLevel.ERROR)
public typealias ClockMark = TimeMark

"Subtracting one TimeMark from another is not a well defined operation because these time marks could have been obtained from the different time sources."

public inline operator fun TimeMark.minus(other: TimeMark): Duration = throw Error("Operation is disallowed.")

public inline operator fun TimeMark.compareTo(other: TimeMark): Int = throw Error("Operation is disallowed.")

private class AdjustedTimeMark(val mark: TimeMark, val adjustment: Duration) : TimeMark() {
    override fun elapsedNow(): Duration = mark.elapsedNow() - adjustment

    override fun plus(duration: Duration): TimeMark = AdjustedTimeMark(mark, adjustment + duration)
}

@sinceKotlin("1.3")
@experimentalTime
@kotlin.internal.InlineOnly
@deprecated("Use TimeSource interface instead.", DeprecationLevel.ERROR)
public typealias Clock = TimeSource

@sinceKotlin("1.3")
@experimentalTime
@kotlin.internal.InlineOnly
@deprecated("Use TimeMark class instead.", DeprecationLevel.ERROR)
public typealias ClockMark = TimeMark

"Comparing one TimeMark to another is not a well defined operation because these time marks could have been obtained from the different time sources."

public inline operator fun TimeMark.compareTo(other: TimeMark): Int = throw Error("Operation is disallowed.")

public inline operator fun TimeMark.minus(other: TimeMark): Duration = throw Error("Operation is disallowed.")

"Using TimeSource interface instead.", DeprecationLevel.ERROR)
public typealias Clock = TimeSource

"Using TimeMark class instead.", DeprecationLevel.ERROR)
public typealias ClockMark = TimeMark

"Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language
Use of this source code is governed by the Apache 2.0 license that can be found in the LICENSE.txt file.

An abstract class used to implement time sources that return their readings as [Long] values in the specified [unit].

This protected method should be overridden to return the current reading of the time source expressed as a [Long] number in the unit specified by the [unit] property. *protected abstract fun read(): Long*

LongTimeMark(private val startedAt: Long, private val timeSource: AbstractLongTimeSource, private val offset: Duration) : TimeMark() {

override fun elapsedNow(): Duration = (timeSource.read() - startedAt).toDuration(timeSource.unit) - offset

override fun plus(duration: Duration): TimeMark = LongTimeMark(startedAt, timeSource, offset + duration)

override fun markNow(): TimeMark = LongTimeMark(read(), this, Duration.ZERO)

A time source that has programmatically updatable readings. It is useful as a predictable source of time in tests.

The current reading value can be advanced by the specified duration amount with the operator [plusAssign]:

```kotlin
val timeSource = TestTimeSource()

timeSource += 10.seconds
```

Implementation note: the current reading value is stored as a [Long] number of nanoseconds, thus it's capable to represent a time range of approximately ±292 years.

Should the reading value overflow as the result of [plusAssign] operation, an [IllegalStateException] is thrown.

`// advances the current reading value of this time source by the specified [duration].

*protected abstract fun read(): Long* advances the current reading value of this time source.

If the duration being added is `0.6.nanoseconds`, the reading doesn't advance because the duration is rounded down to zero nanoseconds.

For example, if the reading is `1000000000000000L`, and the specified duration is 300000000 nanoseconds, the reading value is rounded down to 1000000000000000L.

When the value overflows as the result of this operation, an [IllegalStateException] is thrown.

else when delta fits in long, add it as a `Long`

// when delta fits in long, add it as a `Double`

if (reading xor longDelta >= 0 && reading xor newReading < 0)

newReading = reading + longDelta

if (reading xor longDelta >= 0 && reading xor newReading < 0)
```kotlin
"kotlin.time.AbstractLongTimeSource"), DeprecationLevel.ERROR)/n\public typealias AbstractLongClock =
AbstractLongTimeSource\@SinceKotlin("1.3")\@ExperimentalTime\@Deprecated("Use
AbstractDoubleTimeSource instead.", \ReplaceWith("AbstractDoubleTimeSource\"),
"kotlin.time.AbstractDoubleTimeSource\"), DeprecationLevel.ERROR)/n\public typealias AbstractDoubleClock =
AbstractDoubleTimeSource\@SinceKotlin("1.3")\@ExperimentalTime\@Deprecated("Use TestTimeSource
instead.", \ReplaceWith("TestTimeSource\"), "kotlin.time.TestTimeSource\"), DeprecationLevel.ERROR)/n\public
typealias TestClock = TestTimeSource, */\n*/n * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming
Language contributors.*n * Use of this source code is governed by the Apache 2.0 license that can be found in the
license/LICENSE.txt file.*n */n\package kotlin.time\nimport kotlin.contracts.*

/**
 * Executes the given function [block] and returns the duration of elapsed time interval.*n * The elapsed time is measured with
[TimeSource.Monotonic].*n */n\@SinceKotlin("1.3")\@ExperimentalTime\public inline fun
measureTime(block: () -> Unit): Duration {
    contract {
        callsInPlace(block,
        InvocationKind.EXACTLY_ONCE)\n    }\n    return TimeSource.Monotonic.measureTime(block)\n}

/**
 * Executes the given function [block] and returns an instance of [TimedValue] class, containing both
 * the result of function execution and the duration of elapsed time interval.*n * The elapsed time is measured with
[TimeSource.Monotonic].*n */n\@SinceKotlin("1.3")\@ExperimentalTime\\package
kotlin\nimport kotlin.coroutines.*\nimport kotlin.coroutines.intrinsics.*

/**
 * Defines deep recursive function
that keeps its stack on the heap,
 * which allows very deep recursive computations that do not use the actual call
 * stack.*n * To initiate a call to this deep recursive function use its [invoke] function.*n * As a rule of thumb, it should
 * be used if recursion goes deeper than a thousand calls.*n * The [DeepRecursiveFunction] takes one parameter of
 * type [T] and returns a result of type [R].*n * The [block] of code defines the body of a recursive function. In this
 * block\n * [callRecursive]|DeepRecursiveScope.callRecursive| function can be used to make a recursive call\n * to
 * the declared function. Other instances of [DeepRecursiveFunction] can be called in this scope with
 * `callRecursive` extension, too.*n * For example, take a look at the following recursive tree class and a deeply\n * recursive instance of this tree with 100K nodes:*n *\n * class Tree(val left: Tree? = null, val right: Tree? = null)\n * val deepTree = generateSequence(Tree) { Tree(it) }\n * .take(100_000).last()\n * The [DeepRecursiveFunction] function can be defined to compute a depth of a tree:*n *\n * Int =\n * if (t == null) 0 else max(depth(t.left), depth(t.right)) + 1\n */n\package
kotlin\nimport kotlin.coroutines.coroutines.*n\n\package kotlin.coroutines.intrinsics.*n * Defines deep recursive function
that keeps its stack on the heap,*n * which allows very deep recursive computations that do not use the actual call
 * stack.*n * To initia
```
successfully computes `depth(deepTree)` expression:

```
val depth = DeepRecursiveFunction<Tree?, Int> { t ->
    if (t == null) 0 else max(callRecursive(t.left),
    callRecursive(t.right)) + 1
}
println(depth(deepTree)) // Ok
```

Deep recursive functions can also mutually call each other using a heap for the stack via an `callRecursive()` method. For example, the following pair of mutually recursive functions computes the number of tree nodes at even depth in the tree.

```
val mutualRecursion = object {
    val even: DeepRecursiveFunction<Tree?, Int> = DeepRecursiveFunction { t ->
        if (t == null) 0 else odd.callRecursive(t.left) + odd.callRecursive(t.right) + 1
    }
    val odd: DeepRecursiveFunction<Tree?, Int> = DeepRecursiveFunction { t ->
        if (t == null) 0 else even.callRecursive(t.left) + even.callRecursive(t.right) + 1
    }
}
```

@ExperimentalStdlibApi
public class DeepRecursiveFunction<T, R>(
    internal val block: suspend DeepRecursiveScope<T, R>.(T) -> R
)

/**
 * Initiates a call to this deep recursive function, forming a root of the call tree.
 * This operator should not be used from inside of [DeepRecursiveScope] as it
 * uses the call stack slot for an initial recursive invocation. From inside of [DeepRecursiveScope]
 * use 
 * [callRecursive][DeepRecursiveScope.callRecursive].
 * @param T the function parameter type.
 * @param R the function result type.
 * @param block the function body.
 */
@SinceKotlin("1.4")
@ExperimentalStdlibApi
public operator fun <T, R> DeepRecursiveFunction<T, R>.invoke(value: T): R =
    DeepRecursiveScopeImpl<T, R>(block, value).runCallLoop()
component can have, a constant value 255. /*
// NOTE: Must be placed before CURRENT because its
initialization requires this field being initialized in JS
public const val MAX_COMPONENT_VALUE = 255
/**
 * Returns the current version of the Kotlin standard library.
 */
@kotlin.jvm.JvmField
public val CURRENT: KotlinVersion = KotlinVersionCurrentValue.get()

/**
 * Returns the current version of the Kotlin standard library.
 */
@kotlin.jvm.JvmField
public val CURRENT: KotlinVersion = KotlinVersionCurrentValue.get()

/**
 * Returns the current version of the Kotlin standard library.
 */
@kotlin.jvm.JvmField
public val CURRENT: KotlinVersion = KotlinVersionCurrentValue.get()

/**
 * Returns the current version of the Kotlin standard library.
 */
@kotlin.jvm.JvmField
public val CURRENT: KotlinVersion = KotlinVersionCurrentValue.get()
Counts the number of set bits in the binary representation of this [Int] number.

*\@SinceKotlin('1.4')*\n*\@WasExperimental(ExperimentalStdlibApi::class)*\npublic expect fun

Int.countOneBits(): Int

*\@SinceKotlin('1.4')*\n*\@WasExperimental(ExperimentalStdlibApi::class)*\npublic expect fun

Int.countLeadingZeroBits(): Int

*\@SinceKotlin('1.4')*\n*\@WasExperimental(ExperimentalStdlibApi::class)*\npublic expect fun

Int.countTrailingZeroBits(): Int

*\@SinceKotlin('1.4')*\n*\@WasExperimental(ExperimentalStdlibApi::class)*\npublic expect fun

Int.takeHighestOneBit(): Int

*\@SinceKotlin('1.4')*\n*\@WasExperimental(ExperimentalStdlibApi::class)*\npublic expect fun

Int.takeLowestOneBit(): Int

*\@SinceKotlin('1.4')*\n*\@WasExperimental(ExperimentalStdlibApi::class)*\npublic expect fun

Int.rotateLeft(bitCount: Int): Int

*\@SinceKotlin('1.3')*\n*\@ExperimentalStdlibApi*\npublic expect fun

Int.rotateRight(bitCount: Int): Int

*\@SinceKotlin('1.4')*\n*\@WasExperimental(ExperimentalStdlibApi::class)*\npublic expect fun

Long.countOneBits(): Int

*\@SinceKotlin('1.4')*\n*\@WasExperimental(ExperimentalStdlibApi::class)*\npublic expect fun

Long.countLeadingZeroBits(): Int

*\@SinceKotlin('1.4')*\n*\@WasExperimental(ExperimentalStdlibApi::class)*\npublic expect fun

Long.countTrailingZeroBits(): Int

*\@SinceKotlin('1.4')*\n*\@WasExperimental(ExperimentalStdlibApi::class)*\npublic expect fun

Long.takeHighestOneBit(): Long

*\@SinceKotlin('1.4')*\n*\@WasExperimental(ExperimentalStdlibApi::class)*\npublic expect fun

Long.takeLowestOneBit(): Long

*\@SinceKotlin('1.4')*\n*\@WasExperimental(ExperimentalStdlibApi::class)*\npublic expect fun

Long.rotateLeft(bitCount: Int): Long

*\@SinceKotlin('1.3')*\n*\@ExperimentalStdlibApi*\npublic expect fun

Long.rotateRight(bitCount: Int): Long
specified [bitCount] number of bits.
* The least significant bits pushed out from the right side reenter the number as the most significant bits on the left side.
* Rotating the number right by a negative bit count is the same as rotating it left by the negated bit count.

`number.rotateRight(-n) == number.rotateLeft(n)`

* Rotating by a multiple of [Long.SIZE_BITS] (64) returns the same number, or more generally:

`number.rotateRight(n % 64) == number.rotateRight(n % 64)`

`*number@SinceKotlin("1.3")@ExperimentalStdlibApi

expect fun Long.rotateRight(bitCount: Int): Long

*/
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public expect fun Long.rotateRight(bitCount: Int): Long

*/
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@kotlin.internal.InlineOnly
public inline fun Byte.rotateRight(bitCount: Int): Byte =
    (toInt().shl(8 - (bitCount and 7)) or (toInt() and 0xFF).ushr(bitCount and 7)).toByte()
number having a single bit set in the position of the most significant set bit of this [Short] number, \* or zero, if this number is zero.

```kotlin
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public inline fun Short.takeHighestOneBit(): Short = (toInt() and 0xFFFF).takeHighestOneBit().toShort()
```

Returns a number having a single bit set in the position of the least significant set bit of this [Short] number, \* or zero, if this number is zero.

```kotlin
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public inline fun Short.takeLowestOneBit(): Short = toInt().takeLowestOneBit().toShort()
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public fun Short.rotateLeft(bitCount: Int): Short =
    (toInt().shl(bitCount and 15) or (toInt() and 0xFFFF).ushr(16 - (bitCount and 15))).toShort()
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public fun Short.rotateRight(bitCount: Int): Short =
    (toInt().shl(16 - (bitCount and 15)) or (toInt() and 0xFFFF).ushr(bitCount and 15)).toShort()
```

### Tuple Representation

- Pair:
  - Represents a generic pair of two values.
  - There is no meaning attached to values in this class, it can be used for any purpose.
  - Pair exhibits value semantics, i.e. two pairs are equal if both components are equal.
  - An example of decomposing it into values:
    ```kotlin
    @sample samples.misc.Tuples.pairDestructuring
    ```

- Triple:
  - Represents a triad of values.
  - There is no meaning attached to values in this class, it can be used for any purpose.
  - Triple exhibits value semantics, i.e. two triples are equal if all three components are equal.
  - An example of decomposing it into values:
    ```kotlin
    @sample samples.misc.Tuples.tripleDestructuring
    ```

### Tuple Operations

- Creating tuples:
  - ```kotlin
  A.to(that: B) = Pair(this, that)
  ```
  - ```kotlin
  Pair<A, B> = Pair(A, B)
  ```

- Accessing tuples:
  - ```kotlin
  A.first
  ```
  ```kotlin
  A.second
  ```

- Converting tuples to other types:
  - ```kotlin
  A.toList(): List<T>
  ```
Triple<out A, out B, out C>({\n  public val first: A,
  public val second: B,
  public val third: C
}): Serializable

  /**
   * Returns string representation of the [Triple] including its [first], [second] and [third] values.
   */
  override fun toString(): String = "($first, $second, $third)"

/**
 * Converts this triple into a list.
 */
public fun <T> Triple<T, T, T>.toList(): List<T> =
  listOf(first, second, third)

/**
 * The first element in the progression.
 */
public val first: UInt =
  getProgressionLastElement(start, endInclusive, step)

/**
 * The last element in the progression.
 */
public val last: UInt =
  getProgressionLastElement(start, endInclusive, step)

/**
 * The step of the progression.
 */
public val step: Int =
  step.toInt() // use 2-complement math for negative steps

override fun iterator(): UIntIterator =
  UIntProgressionIterator(first, last, step)

package kotlin.ranges

import kotlin.internal.*

/**
 * A range of values of type `ULong`.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public class ULongRange(start: ULong, endInclusive: ULong) : ULongProgression(start, endInclusive, 1), ClosedRange<ULong> {
    override val start: ULong get() = first
    override val endInclusive: ULong get() = last

    override fun contains(value: ULong): Boolean = first <= value && value <= last

    override fun isEmpty(): Boolean = if (step > 0) first > last else first < last

    override fun equals(other: Any?): Boolean =
        other is ULongRange && (isEmpty() && other.isEmpty() ||
                first == other.first && last == other.last)

    override fun hashCode(): Int =
        if (isEmpty()) -1 else (31 * (first xor (first shr 32)).toInt() + (last xor (last shr 32)).toInt())

    override fun toString(): String = "$first..$last"

    companion object {

        /** An empty range of values of type ULong. */
        public val EMPTY: ULongRange = ULongRange(ULong.MAX_VALUE, ULong.MIN_VALUE)
    }
}

/**
 * A progression of values of type `ULong`.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public open class ULongProgression
internal constructor(
    start: ULong,
    endInclusive: ULong,
    step: Long
) : Iterable<ULong> {
    init {
        if (step == 0.toLong()) throw kotlin.IllegalArgumentException("Step must be non-zero.\")
        if (step == Long.MIN_VALUE) throw kotlin.IllegalArgumentException("Step must be greater than Long.MIN_VALUE to avoid overflow on negation.\")
    }

    /**
     * The first element in the progression.
     */
    public val first: ULong = start

    /**
     * The last element in the progression.
     */

    public val last: ULong = getProgressionLastElement(start, endInclusive, step)

    /**
     * The step of the progression.
     */
    public val step: Long = step

    override fun iterator(): ULongIterator = ULongProgressionIterator(first, last, step)

    /** Checks if the progression is empty. */
    public open fun isEmpty(): Boolean = if (step > 0) first > last else first < last

    override fun equals(other: Any?): Boolean =
        other is ULongProgression && (isEmpty() && other.isEmpty() ||
                first == other.first && last == other.last)

    override fun hashCode(): Int =
        if (isEmpty()) -1 else (31 * (first xor (first shr 32)).toInt() + (last xor (last shr 32)).toInt() + (step xor (step ushr 32)).toInt())

    override fun toString(): String = if (step > 0) "$first..$last step $step" else "$first downTo $last step ${-step}"
Returns the smaller of two values.

`min(a: UInt, b: UInt): UInt` Returns the greater of two values.

`min(a: ULong, b: ULong): ULong` Returns the greater of two values.

`max(a: UInt, b: UInt): UInt` Returns the greater of two values.

`max(a: ULong, b: ULong): ULong` Returns the greater of two values.

`countOneBits(): Int` Counts the number of set bits in the binary representation of this [UInt] number.

`countLeadingZeroBits(): Int` Counts the number of consecutive most significant bits that are zero in the binary representation of this [UInt] number.

`countTrailingZeroBits(): Int` Counts the number of consecutive least significant bits that are zero in the binary representation of this [UInt] number.

`takeHighestOneBit(): UInt` Returns a number having a single bit set in the position of the most significant set bit of this [UInt] number.

`takeLowestOneBit(): UInt` Returns a number having a single bit set in the position of the least significant set bit of this [UInt] number.

`rotateLeft(bitCount: Int): UInt` Rotates the binary representation of this [UInt] number left by the specified [bitCount] number of bits.

`rotateRight(bitCount: Int): UInt` Rotates the binary representation of this [UInt] number right by the specified [bitCount] number of bits.

The most significant bits pushed out from the right side reenter the number as the least significant bits on the right side. Rotating the number left by a negative bit count is the same as rotating it right by the negated bit count. Rotating by a multiple of [UInt.SIZE_BITS] (32) returns the same number, or more generally the negated bit count.

`rotateLeft(-n) == number.rotateRight(n)` Rotating by a multiple of [UInt.SIZE_BITS] (32) returns the same number, or more generally the negated bit count.
@kotlin.internal.InlineOnly
@kotlin.internal.InlineOnly
public inline fun ULong.countOneBits(): Int = toLong().countOneBits() * Counts the number of consecutive most significant bits that are zero in the binary representation of this [ULong] number.

*\@SinceKotlin("1.4")\@WasExperimental(ExperimentalStdlibApi::class)@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULong.countLeadingZeroBits(): Int = toLong().countLeadingZeroBits() * Counts the number of consecutive least significant bits that are zero in the binary representation of this [ULong] number.

*\@SinceKotlin("1.4")\@WasExperimental(ExperimentalStdlibApi::class)@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULong.countTrailingZeroBits(): Int = toLong().countTrailingZeroBits() * Returns a number having a single bit set in the position of the most significant set bit of this [ULong] number,\n
*\@SinceKotlin("1.4")\@WasExperimental(ExperimentalStdlibApi::class)@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULong.takeHighestOneBit(): ULong = toLong().takeHighestOneBit().toULong() * Returns a number having a single bit set in the position of the least significant set bit of this [ULong] number,\n
*\@SinceKotlin("1.4")\@WasExperimental(ExperimentalStdlibApi::class)@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULong.takeLowestOneBit(): ULong = toLong().takeLowestOneBit().toULong() * Rotates the binary representation of this [ULong] number left by the specified [bitCount] number of bits.\n
*\@SinceKotlin("1.4")\@WasExperimental(ExperimentalStdlibApi::class)@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULong.rotateLeft(bitCount: Int): ULong = toLong().rotateLeft(bitCount).toULong() * Rotates the number left by a negative bit count as the same as rotating it right by the negated bit count.\n
*\@SinceKotlin("1.4")\@WasExperimental(ExperimentalStdlibApi::class)@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByte.countOneBits(): Int = toUInt().countOneBits() * Counts the number of set bits in the binary representation of this [UByte] number.

*\@SinceKotlin("1.4")\@WasExperimental(ExperimentalStdlibApi::class)@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByte.countLeadingZeroBits(): Int = toByte().countLeadingZeroBits() * Counts the number of consecutive least significant bits that are zero in the binary representation of this [UByte] number.

*\@SinceKotlin("1.4")\@WasExperimental(ExperimentalStdlibApi::class)@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByte.countTrailingZeroBits(): Int = toByte().countTrailingZeroBits() * Returns a number having a single bit set in the position of the most significant set bit of this [UByte] number,\n
*\@SinceKotlin("1.4")\@WasExperimental(ExperimentalStdlibApi::class)@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByte.takeHighestOneBit(): UByte = toInt().takeHighestOneBit().toUByte() * Returns a number having a single bit set in the position of the least
significant set bit of this [UByte] number,\n * or zero, if this number is zero.\n
\n
 *\n
 */
@SinceKotlin("1.4")\n@WasExperimental(ExperimentalStdlibApi::class)\n@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly\n
public inline fun UByte.takeLowestOneBit(): UByte =
toInt().takeLowestOneBit().toUByte()\n\n
/**
 * Rotates the number left by a negative bit count is the same as rotating it right by the negated bit count:\n * `number.rotateLeft(-n) == number.rotateRight(n)`\n * Rotating by a multiple of [UByte.SIZE_BITS] (8) returns the same number, or more generally:\n * \n
 */
@SinceKotlin("1.3")\n@ExperimentalStdlibApi\n@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly\n
public inline fun UByte.rotateLeft(bitCount: Int): UByte = toByte().rotateLeft(bitCount).toUByte()\n\n
/**
 * Rotates the number right by a negative bit count is the same as rotating it left by the negated bit count:\n * `number.rotateRight(-n) == number.rotateLeft(n)`\n * Rotating by a multiple of [UShort.SIZE_BITS] (16) returns the same number, or more generally:\n * Counts the number of set bits in the binary representation of this [UShort] number:\n
 */
@SinceKotlin("1.4")\n@WasExperimental(ExperimentalStdlibApi::class)\n@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly\n
public inline fun UShort.countOneBits(): Int = toUInt().countOneBits()\n\n
/**
 * Counts the number of consecutive most significant bits that are zero in the binary representation of this [UShort] number:\n
 */
@SinceKotlin("1.4")\n@WasExperimental(ExperimentalStdlibApi::class)\n@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly\n
public inline fun UShort.countLeadingZeroBits(): Int = toShort().countLeadingZeroBits()\n\n
/**
 * Counts the number of consecutive least significant bits that are zero in the binary representation of this [UShort] number:\n
 */
@SinceKotlin("1.4")\n@WasExperimental(ExperimentalStdlibApi::class)\n@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly\n
public inline fun UShort.countTrailingZeroBits(): Int = toShort().countTrailingZeroBits()\n\n
/**
 * Returns a number having a single bit set in the position of the most significant set bit of this [UShort] number,\n * or zero, if this number is zero.\n
 */
@SinceKotlin("1.4")\n@WasExperimental(ExperimentalStdlibApi::class)\n@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly\n
public inline fun UShort.takeHighestOneBit(): UShort =
toInt().takeHighestOneBit().toUShort()\n\n
/**
 * Rotates the binary representation of this [UShort] number left by the specified [bitCount] number of bits.\n * The most significant bits pushed out from the left side reenter the number as the least significant bits on the right side:\n * Rotating by a multiple of [UShort.SIZE_BITS] (16) returns the same number, or more generally:\n * \n
 */
@SinceKotlin("1.3")\n@ExperimentalStdlibApi\n@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly\n
public inline fun UShort.rotateLeft(bitCount: Int): UShort = toShort().rotateLeft(bitCount).toUShort()\n\n
/**
 * Rotates the binary representation of this [UShort] number right by the specified [bitCount] number of bits.\n * The least significant bits pushed out from the right side reenter the number as the most significant bits on the left side:\n * Rotating by a negative bit count is the same as rotating it left by the negated bit count:\n * 

 */
@SinceKotlin("1.4")\n@WasExperimental(ExperimentalStdlibApi::class)\n@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly\n
public inline fun UShort.rotateRight(bitCount: Int): UShort = toShort().rotateRight(bitCount).toUShort()
number.rotateRight(-n) == number.rotateLeft(n)

* number.rotateRight(n) == number.rotateRight(n % 16)

* @SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes

* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

* @PublishedApi
* @file:OptIn(ExperimentalUnsignedTypes::class)
* @package kotlin.internal

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

@file:kotlin.jvm.JvmName("UStringsKt")
package kotlin.text

/**
 * Returns a string representation of this [Byte] value in the specified [radix].
 *
 * @throws IllegalArgumentException when [radix] is not a valid radix for number to string conversion.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
//@kotlin.internal.InlineOnly
public /*inline*/ fun UByte.toString(radix: Int): String = this.toInt().toString(radix)

/**
 * Returns a string representation of this [Short] value in the specified [radix].
 *
 * @throws IllegalArgumentException when [radix] is not a valid radix for number to string conversion.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
//@kotlin.internal.InlineOnly
public /*inline*/ fun UShort.toString(radix: Int): String = this.toInt().toString(radix)

/**
 * Returns a string representation of this [Int] value in the specified [radix].
 *
 * @throws IllegalArgumentException when [radix] is not a valid radix for number to string conversion.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
//@kotlin.internal.InlineOnly
public fun ULong.toString(radix: Int): String = ulongToString(this.toLong(), checkRadix(radix))

/**
 * Parses the string

`number.rotateRight(-n) == number.rotateLeft(n)`

Rotating by a multiple of `UShort.SIZE_BITS` (16) returns the same number, or more generally `number.rotateRight(n) == number.rotateRight(n % 16)`.
as a signed [UByte] number and returns the result. * @throws NumberFormatException if the string is not a valid representation of a number. * @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun String.toUByteOrNull(): UByte? {
    val int = this.toUIntOrNull(radix) ?: return null
    if (int > UByte.MAX_VALUE) return null
    return int.toUByte()
}

/**
 * Parses the string as an [UShort] number and returns the result. * @throws NumberFormatException if the string is not a valid representation of a number. * @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun String.toUShortOrNull(): UShort? {
    val int = this.toUIntOrNull(radix) ?: return null
    if (int > UShort.MAX_VALUE) return null
    return int.toUShort()
}

/**
 * Parses the string as an [UInt] number and returns the result. * @throws NumberFormatException if the string is not a valid representation of a number. * @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun String.toUIntOrNull(): UInt? {
    checkRadix(radix)

    val length = this.length
    if (length == 0)
return null

val limit: UInt = UInt.MAX_VALUE
val start: Int = 0

if (firstChar < '0') { // if (length == 1 || firstChar != '+') return null
    start = 1
} else {
    if (start == 0) { // return result
        result += digit.toUInt()
        return result
    } else {
        return result
    }
}

// Get the first character.

val firstChar = this[0]
if (firstChar < '0') {
    if (length == 1 || firstChar != '+') return null
    start = 1
} else {
    start = 0
}

val limitForMaxRadix = 119304647u  // limit / 36
val uradix = radix.toUInt()

val result = 0u
for (i in start until length) {
    val digit = digitOf(this[i], radix)
    if (digit < 0) return null
    if (result > limitBeforeMul) {
        if (limitBeforeMul == limitForMaxRadix) {
            limitBeforeMul = limit / uradix
        } else {
            return null
        }
    }
    result *= uradix
    val beforeAdding = result
    result += digit.toUInt()
    if (result < beforeAdding) return null // overflow has happened
}

return result

/**
 * Parses the string as an [ULong] number and returns the result
 * or `null` if the string is not a valid representation of a number.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun String.toULongOrNull(): ULong? = toULongOrNull(radix = 10)

/**
 * Parses the string as an [ULong] number and returns the result
 * or `null` if the string is not a valid representation of a number.
 *
 * @throws
 * IllegalArgumentException when [radix] is not a valid radix for string to number conversion.

 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */
@file:Suppress("INVISIBLE_REFERENCE", "INVISIBLE_MEMBER")
@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("MathKt")

package kotlin.math

// constants, can't use them from nativeMath as they are not constants there

/** Ratio of the circumference of a circle to its diameter, approximately 3.14159. */
@SinceKotlin("1.2")
public const val PI: Double = 3.141592653589793
/** Base of the natural logarithms, approximately 2.71828. */
@SinceKotlin("1.2")
public const val E: Double = 2.718281828459045

// region MathInt ========================================

/** Computes the sine of the angle [x] given in radians.
 * Special cases:
 * - `sin(NA) + Inf/Inf)` is `NA`
 */
@SinceKotlin("1.2")
public
expect fun sin(x: Double): Double

```java
public expect fun cos(x: Double):
```
*\@SinceKotlin("1.2")*

public expect fun log(x: Double, base: Double): Double\n
*\@SinceKotlin("1.2")*

public expect fun log(x: Double): Double

*\@SinceKotlin("1.2")*

public expect fun ln(x: Double): Double

*\@SinceKotlin("1.2")*

public expect fun log10(x: Double): Double

*\@SinceKotlin("1.2")*

public expect fun log2(x: Double): Double

*\@SinceKotlin("1.2")*

public expect fun ln1p(x: Double): Double

*\@SinceKotlin("1.2")*

public expect fun ceil(x: Double): Double

*\@SinceKotlin("1.2")*

public expect fun floor(x: Double): Double

*\@SinceKotlin("1.2")*

public expect fun truncate(x: Double): Double

*\@SinceKotlin("1.2")*

public expect fun round(x: Double): Double

*\@SinceKotlin("1.2")*

public expect fun abs(x: Double): Double

*\@SinceKotlin("1.2")*

public expect fun sign(x: Double): Double

*\@SinceKotlin("1.2")*

public expect fun min(a: Double, b: Double): Double

*\@SinceKotlin("1.2")*

public expect fun max(a: Double, b: Double): Double

*\@SinceKotlin("1.2")*

public expect fun pow(x: Double): Double

*\@SinceKotlin("1.2")*

public expect fun pow(n: Int): Double

*\@SinceKotlin("1.2")*

public expect val absoluteValue: Double

*\@SinceKotlin("1.2")*

public expect val sign: Double

*\@SinceKotlin("1.2")*

public expect fun withSign(sign: Double): Double

*\@SinceKotlin("1.2")*

public expect fun withSign(sign: Int): Double

*\@SinceKotlin("1.2")*

public expect val ulp: Double

*\@SinceKotlin("1.2")*

public expect val MIN_VALUE: Double

*\@SinceKotlin("1.2")*

public expect val MAX_VALUE: Double

*\@SinceKotlin("1.2")*

public expect val MAX_ulp: Double

*\@SinceKotlin("1.2")*

public expect val isFinite: Boolean

*\@SinceKotlin("1.2")*

public expect val isInfinite: Boolean

*\@SinceKotlin("1.2")*

public expect val isNaN: Boolean

*\@SinceKotlin("1.2")*

public expect val isNegative: Boolean

*\@SinceKotlin("1.2")*

public expect val isZero: Boolean
this value in direction of positive infinity.

```kotlin
public expect fun Double.nextUp(): Double
```

```kotlin
public expect fun Double.nextDown(): Double
```

```kotlin
public expect fun Double.nextTowards(to: Double): Double
```

```kotlin
public expect fun Double.roundToInt(): Int
```

```kotlin
public expect fun Double.roundToLong(): Long
```

```kotlin
// endregion
```

```kotlin
// region ================ Float Math
```

```kotlin
public expect fun sin(x: Float): Float
```

```kotlin
public expect fun cos(x: Float): Float
```

```kotlin
public expect fun tan(x: Float): Float
```

```kotlin
public expect fun asin(x: Float): Float
```

```kotlin
public expect fun acos(x: Float): Float
```

```kotlin
public expect fun atan(x: Float): Float
```

```kotlin
public expect fun atan2(y: Float, x: Float): Float
```

```kotlin
public expect fun sinh(x: Float): Float
```

```kotlin
public expect fun cosh(x: Float): Float
```

```kotlin
public expect fun tanh(x: Float): Float
```

```kotlin
public expect fun asinh(x: Float): Float
```

```kotlin
public expect fun acosh(x: Float): Float
```

```kotlin
public expect fun @SinceKotlin("1.2")
```

```kotlin
// endregion
```

```kotlin
// OpenSource Used In security_automation identity-guest-cs
```

```kotlin
1810
```
Open Source Used In security_automation identity-guest-cs 1811
Open Source Used In security_automation identity-guest-cs 1812
Open Source Used in security_automation identity-guest-cs 1815
Open Source Used in security_automation identity-guest-cs 1852
Open Source Used in security_automation identity-guest-cs 1856
Open Source Used in security_automation identity-guest-cs  1870
Open Source Used In security_automation identity-guest-cs 1927
Open Source Used In security_automation identity-guest-cs  1982
Open Source Used in security_automation identity-guest-cs 2028
1.185 listenablefuture 1.0
1.185.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2007 The Guava Authors
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/cola/permits/1258121121_1643033053.9/0/listenablefuture-1-0-sources-jar/com/google/common/util/concurrent/ListenableFuture.java

1.186 cloning 1.9.0
1.186.1 Available under license:

No license file was found, but licenses were detected in source scan.

<name>Apache License, Version 2.0</name>
<url>http://www.apache.org/licenses/LICENSE-2.0</url>

Found in path(s):
* /opt/cola/permits/1135840428_1613612967.07/0/cloning-1-9-0-jar/META-INF/maven/uk.com.robust-it/cloning/pom.xml

No license file was found, but licenses were detected in source scan.

Manifest-Version: 1.0
Bnd-LastModified: 1349353954479
Build-Jdk: 1.7.0_07
Built-By: ariskk
Bundle-Description: The cloning library is a small, open source (Apache licensed) Java library which deep-clones objects. The objects do not have to implement the Cloneable interface. Effectively, this library can clone ANY Java object. It can be used i.e. in cache implementations, if you don't want the cached object to be modified or whenever you
want to create a deep copy of an object. Sample usage: You can create a single instance of Cloner and use it throughout your application. i.e. Cloner cloner = new Cloner();...cloner.deepClone(o); Please check http://robust-it.co.uk/clone/ for more examples (spring integration, immutable objects etc).

Bundle-License: http://www.apache.org/licenses/LICENSE-2.0
Bundle-ManifestVersion: 2
Bundle-Name: cloning
Bundle-SymbolicName: uk.com.robust-it.cloning
Bundle-Version: 1.9.0
Created-By: Apache Maven Bundle Plugin
Export-Package: com.rits.cloning;uses:="org.objenesis";version="1.9.0",com.rits.perspectives;uses:="com.rits.cloning";version="1.9.0"
Import-Package: org.objenesis;version="[1.2,2)"
Tool: Bnd-1.50.0

Found in path(s):
* /opt/cola/permits/1135840428_1613612967.07/0/cloning-1.9-0.jar/META-INF/MANIFEST.MF

1.187 lombok 1.18.16
1.187.1 Available under license:
Copyright (C) 2009-2020 The Project Lombok Authors.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.188 spring-integration-core 5.4.11
1.188.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014-2019 the original author or authors.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/config/EnablePublisher.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/config/IntegrationConfigurationInitializer.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/config/annotation/BridgeFromAnnotationPostProcessor.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/util/CompositeExecutor.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/scheduling/PollSkipStrategy.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/annotation/InboundChannelAdapter.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/config/IntegrationConverterInitializer.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/config/IntegrationConfigurationBeanFactoryPostProcessor.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/config/GlobalChannelInterceptorInitializer.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/configuration/config/IntegrationRegistrar.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/channel/interceptor/VetoCapableInterceptor.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2020-2021 the original author or authors.
 */
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/util/IntegrationReactiveUtils.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2017-2020 the original author or authors.
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/channel/MessageChannelReactiveUtils.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/support/json/JacksonPresent.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/support/json/EmbeddedJsonHeadersMessageMapper.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/handler/ReplyProducingMessageHandlerWrapper.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/dsl/FluxMessageChannelSpec.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/gateway/AnnotationGatewayProxyFactoryBean.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/json/SimpleJsonSerializer.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/StaticMessageHeaderAccessor.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2013-2019 the original author or authors.
 * 
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 * https://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/config/xml/SpelPropertyAccessorsParser.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/util/StackTraceUtils.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/json/JsonPropertyAccessor.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/config/xml/SpelFunctionParser.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/IntegrationMessageHeaderAccessor.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/support/IdGenerators.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/transformer/support/MessageProcessingHeaderValueMessageProcessor.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/transformer/support/HeaderValueMessageProcessor.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/transformer/support/AbstractHeaderValueMessageProcessor.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/gateway/MethodArgsMessageMapper.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-
/*
 * Copyright 2019-2021 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *      https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/dsl/BaseIntegrationFlowDefinition.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2018-2019 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *      https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
/*
 * Copyright 2016-2019 the original author or authors.
 */
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*    https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1265673586_1643922104.6/spring-integration-core-5.4.11-sources.jar/org/springframework/integration/endpoint/AbstractFetchLimitingMessageSource.java
* /opt/cola/permits/1265673586_1643922104.6/spring-integration-core-5.4.11-sources.jar/org/springframework/integration/support/MapBuilder.java
* /opt/cola/permits/1265673586_1643922104.6/spring-integration-core-5.4.11-sources.jar/org/springframework/integration/dsl/IntegrationComponentSpec.java
* /opt/cola/permits/1265673586_1643922104.6/spring-integration-core-5.4.11-sources.jar/org/springframework/integration/graph/CompositeMessageHandlerNode.java
* /opt/cola/permits/1265673586_1643922104.6/spring-integration-core-5.4.11-sources.jar/org/springframework/integration/dsl/IntegrationFlowBuilder.java
* /opt/cola/permits/1265673586_1643922104.6/spring-integration-core-5.4.11-sources.jar/org/springframework/integration/dsl/MessageSourceSpec.java
* /opt/cola/permits/1265673586_1643922104.6/spring-integration-core-5.4.11-sources.jar/org/springframework/integration/handler/advice/AbstractHandleMessageAdvice.java
* /opt/cola/permits/1265673586_1643922104.6/spring-integration-core-5.4.11-sources.jar/org/springframework/integration/graph/ErrorCapableDiscardingMessageHandlerNode.java
* /opt/cola/permits/1265673586_1643922104.6/spring-integration-core-5.4.11-sources.jar/org/springframework/integration/context/ExpressionCapable.java
* /opt/cola/permits/1265673586_1643922104.6/spring-integration-core-5.4.11-sources.jar/org/springframework/integration/handler/advice/HandleMessageAdvice.java
* /opt/cola/permits/1265673586_1643922104.6/spring-integration-core-5.4.11-sources.jar/org/springframework/integration/dsl/WireTapSpec.java
* /opt/cola/permits/1265673586_1643922104.6/spring-integration-core-5.4.11-sources.jar/org/springframework/integration/endpoint/IntegrationConsumer.java
* /opt/cola/permits/1265673586_1643922104.6/spring-integration-core-5.4.11-sources.jar/org/springframework/integration/support/management/SubscribableChannelManagement.java
* /opt/cola/permits/1265673586_1643922104.6/spring-integration-core-5.4.11-sources.jar/org/springframework/integration/handler/LambdaMessageProcessor.java
* /opt/cola/permits/1265673586_1643922104.6/spring-integration-core-5.4.11-sources.jar/org/springframework/integration/support/StringStringMapBuilder.java
* /opt/cola/permits/1265673586_1643922104.6/spring-integration-core-5.4.11-sources.jar/org/springframework/integration/transformer/StreamTransformer.java
* /opt/cola/permits/1265673586_1643922104.6/spring-integration-core-5.4.11-sources.jar/org/springframework/integration/dsl/ComponentsRegistration.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2017-2019 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *      https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/graph/LinkNode.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/graph/ErrorCapableCompositeMessageHandlerNode.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/graph/ErrorCapableNode.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/exception/(ErrorMessageStrategy.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/exception/MessagingExceptionWrapper.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/dsl/MessageProcessorSpec.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/dsl/PollerFactory.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/dsl/CorrelationHandlerSpec.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/dsl/MessagingGatewaySpec.java

Open Source Used In security_automation identity-guest-cs 2046
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2007-2019 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/store/MessageStoreException.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2020 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/support/locks/RenewableLockRegistry.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/dsl/ReactiveMessageHandlerSpec.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/aop/ReceiveMessageAdvice.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/handler/ReactiveMessageHandlerAdapter.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2016-2021 the original author or authors.
*/
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Found in path(s):
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/dsl/Channels.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/dsl/MessageChannels.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/dsl/PublishSubscribeChannelSpec.java

No license file was found, but licenses were detected in source scan.

Copyright 2015-2019 the original author or authors.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Found in path(s):
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/handler/MessageTriggerAction.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/support/converter/PassThruMessageConverter.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/leader/event/OnGrantedEvent.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/transformer/EncodingPayloadTransformer.java
You may obtain a copy of the License at
* https://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/config/annotation/AnnotationMetadataAdapter.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/support/channel/ChannelResolverUtils.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/util/JavaUtils.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/channel/ChannelUtils.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/transformer/SimpleToAvroTransformer.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/transformer/support/AvroHeaders.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/aggregator/DefaultAggregateHeadersFunction.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/handler/advice/CacheRequestHandlerAdvice.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/graph/ReceiveCountersAware.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/graph/TimerStats.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/graph/IntegrationPattern.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/graph/SendTimers.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/graph/PollableChannelNode.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/graph/MicrometerNodeEnhancer.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/graph/ReceiveCounters.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/graph/SendTimersAware.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/handler/advice/RateLimiterRequestHandlerAdvice.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/IntegrationPatternType.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015-2020 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-
  jar/org/springframework/integration/transformer/SimpleFromAvroTransformer.java

No license file was found, but licenses were detected in source scan.
Found in path(s):
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/config/xml/BarrierParser.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/codec/kryo/AbstractKryoRegistrar.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/codec/kryo/AbstractKryoCodec.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/codec/kryo/MessageKryoRegistrar.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/codec/kryo/KryoClassMapRegistrar.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/channel/FluxMessageChannel.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/codec/kryo/CompositeKryoRegistrar.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/config/xml/TransactionSynchronizationFactoryParser.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/support/json/AbstractJsonInboundMessageMapper.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/aop/MethodAnnotationPublisherMetadataSource.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/support/management/IntegrationManagement.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2019 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/handler/ExpressionEvaluatingMessageProcessor.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/config/xml/TransactionSynchronizationFactoryParser.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/support/json/AbstractJsonInboundMessageMapper.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/aop/MethodAnnotationPublisherMetadataSource.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/config/xml/ResequencerParser.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/support/locks/LockRegistry.java

* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/transformer/AbstractPayloadTransformer.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/router/MethodInvokingRouter.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/channel/RendezvousChannel.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/selector/MessageSelectorChain.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/util/WhileLockedProcessor.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/core/MessageProducer.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/aggregator/MessageCountReleaseStrategy.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/config/xml/ErrorMessageExceptionTypeRouterParser.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/selector/UnexpiredMessageSelector.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/aop/MethodNameMappingPublisherMetadataSource.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/dispatcher/UnicastingDispatcher.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2014-2021 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* *
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/annotation/MessagingGateway.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/context/IntegrationProperties.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2016-2020 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* *
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
No license file was found, but licenses were detected in source scan.
/*
* Copyright 2002-2020 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*    https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1265673586_1643922104.6/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/transformer/ContentEnricher.java
* /opt/cola/permits/1265673586_1643922104.6/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/util/AbstractExpressionEvaluator.java
* /opt/cola/permits/1265673586_1643922104.6/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/aop/PublisherAnnotationAdvisor.java
* /opt/cola/permits/1265673586_1643922104.6/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/channel/AbstractMessageChannel.java
* /opt/cola/permits/1265673586_1643922104.6/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/filter/MessageFilter.java
* /opt/cola/permits/1265673586_1643922104.6/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/gateway/MessagingGatewaySupport.java
* /opt/cola/permits/1265673586_1643922104.6/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/config/xml/DefaultOutboundChannelAdapterParser.java
* /opt/cola/permits/1265673586_1643922104.6/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/handler/MethodInvokingMessageHandler.java
* /opt/cola/permits/1265673586_1643922104.6/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/router/AbstractMessageProcessingRouter.java
* /opt/cola/permits/1265673586_1643922104.6/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/support/MessageBuilder.java
* /opt/cola/permits/1265673586_1643922104.6/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/aggregator/AbstractCorrelatingMessageHandler.java
* /opt/cola/permits/1265673586_1643922104.6/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/dispatcher/RoundRobinLoadBalancingStrategy.java
* /opt/cola/permits/1265673586_1643922104.6/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/handler/advice/RequestHandlerRetryAdvice.java
* /opt/cola/permits/1265673586_1643922104.6/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/config/annotation/ServiceActivatorAnnotationPostProcessor.java
* /opt/cola/permits/1265673586_1643922104.6/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/metadata/MetadataStore.java
* /opt/cola/permits/1265673586_1643922104.6/spring-integration-core-5-4-11-sources-
jar/org/springframework/integration/channel/AbstractPollableChannel.java

* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-
jar/org/springframework/integration/config/xml/SelectorChainParser.java

* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-
jar/org/springframework/integration/handler/ServiceActivatingHandler.java

* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-
jar/org/springframework/integration/channel/QueueChannel.java

* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-
jar/org/springframework/integration/config/xml/ApplicationEventMulticasterParser.java

* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-
jar/org/springframework/integration/json/JsonToObjectTransformer.java

* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-
jar/org/springframework/integration/router/AbstractMappingMessageRouter.java

* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-
jar/org/springframework/integration/history/MessageHistoryConfigurer.java

* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-
jar/org/springframework/integration/config/ConsumerEndpointFactoryBean.java

* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-
jar/org/springframework/integration/router/RecipientListRouter.java

* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-
jar/org/springframework/integration/handler/support/MessagingMethodInvokerHelper.java

* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-
jar/org/springframework/integration/gateway/GatewayMethodInboundMessageMapper.java

* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-
jar/org/springframework/integration/util/BeanFactoryTypeConverter.java

* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-
jar/org/springframework/integration/handler/advice/ExpressionEvaluatingRequestHandlerAdvice.java

* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-
jar/org/springframework/integration/aggregator/MethodInvokingReleaseStrategy.java

* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-
jar/org/springframework/integration/dispatcher/OrderedAwareCopyOnWriteArraySet.java

* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-
jar/org/springframework/integration/transformer/SyslogToMapTransformer.java

* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-
jar/org/springframework/integration/endpoint/SourcePollingChannelAdapter.java

* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-
jar/org/springframework/integration/endpoint/MethodInvokingMessageSource.java

* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-
jar/org/springframework/integration/aggregator/SequenceSizeReleaseStrategy.java

* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-
jar/org/springframework/integration/config/xml/DefaultInboundChannelAdapterParser.java

* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-
jar/org/springframework/integration/config/AbstractSimpleMessageHandlerFactoryBean.java

* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-
jar/org/springframework/integration/handler/AbstractReplyProducingMessageHandler.java

* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-
jar/org/springframework/integration/config/AbstractStandardMessageHandlerFactoryBean.java

* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-
jar/org/springframework/integration/config/xml/JsonToObjectTransformerParser.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-
jar/org/springframework/integration/endpoint/MessageProducerSupport.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-
jar/org/springframework/integration/config/GlobalChannelInterceptorProcessor.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-
jar/org/springframework/integration/transformer/AbstractMessageProcessingTransformer.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-
jar/org/springframework/integration/config/xml/IntegrationNamespaceHandler.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-
jar/org/springframework/integration/config/IdGeneratorConfigurer.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2021 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *      https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-
jar/org/springframework/integration/channel/NullChannel.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-
jar/org/springframework/integration/config/SourcePollingChannelAdapterFactoryBean.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-
jar/org/springframework/integration/config/xml/PublishSubscribeChannelParser.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-
jar/org/springframework/integration/config/DefaultConfiguringBeanFactoryPostProcessor.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-
jar/org/springframework/integration/channel/PublishSubscribeChannel.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-
jar/org/springframework/integration/endpoint/history/MessageHistory.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-
jar/org/springframework/integration/endpoint/AbstractPollingEndpoint.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-
jar/org/springframework/integration/support/json/JacksonJsonUtils.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-
jar/org/springframework/integration/endpoint/AbstractEndpoint.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2019-2020 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
*        https://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/aggregator/FluxAggregatorMessageHandler.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/aggregator/DelegatingMessageGroupProcessor.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/handler/MessageHandlerSupport.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/handler/AbstractReactiveMessageHandler.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/dsl/GatewayProxySpec.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2013-2020 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
*        https://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/support/json/AbstractJacksonJsonObjectMapper.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/events/IntegrationEvent.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/transformer/support/ExpressionEvaluatingHeaderValueMessageProcessor.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/channel/DefaultHeaderChannelRegistry.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/support/json/JsonObjectMapper.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/mapping/support/JsonHeaders.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/config/xml/IdempotentReceiverInterceptorParser.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/selector/MetadataStoreSelector.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/scheduling/PollSkipAdvice.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/channel/FixedSubscriberChannel.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/handler/advice/IdempotentReceiverInterceptor.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/config/xml/ScatterGatherParser.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/support/MutableMessage.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/transformer/support/ExpressionEvaluatingHeaderValueMessageProcessor.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/support/json/AbstractJacksonJsonObjectMapper.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/support/json/JsonObjectMapper.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/config/xml/ScatterGatherParser.java

No license file was found, but licenses were detected in source scan.

/**
 * Copyright 2014-2020 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/config/xml/IdempotentReceiverInterceptorParser.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/selector/MetadataStoreSelector.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/config/xml/ScatterGatherParser.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/channel/DefaultHeaderChannelRegistry.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/transformer/support/ExpressionEvaluatingHeaderValueMessageProcessor.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/scheduling/PollSkipAdvice.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/channel/FixedSubscriberChannel.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/handler/advice/IdempotentReceiverInterceptor.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/support/MutableMessage.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/config/xml/ScatterGatherParser.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/support/json/AbstractJacksonJsonObjectMapper.java
* /opt/cola/permits/1265673586_1643922104.6/0/spring-integration-core-5-4-11-sources-jar/org/springframework/integration/support/json/AbstractJacksonJsonObjectMapper.java
1.189 istack-commons 3.0.5
1.189.1 Available under license:
Found license 'Common Development and Distribution License (CDDL) 1.1' in 'Copyright (c) 1997-2017 Oracle and/or its affiliates. All rights reserved. The contents of this file are subject to the terms of either the GNU General Public License Version 2 only ("GPL") or the Common Development and Distribution License("CDDL") (collectively, the "License"). You may not use this file except in compliance with the License. You can Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the GPL Version 2 section of the License'

Found license 'General Public License 2.0' in 'Copyright (c) 1997-2017 Oracle and/or its affiliates. All rights reserved. The contents of this file are subject to the terms of either the GNU General Public License Version 2 only ("GPL") or the Common Development and Distribution License("CDDL") (collectively, the "License"). You may not use this file except in compliance with the License. You can Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the GPL Version 2 section of the License'

1.190 jnr-posix 3.1.2
1.190.1 Available under license:
Found license 'GNU Lesser General Public License' in '* License Version 2.0 (the "License"); you may not use this file * except in compliance with the License. You may obtain a copy of * Software distributed under the License is distributed on an "AS * Alternatively, the contents of this file may be used under the terms of * either of the GNU General Public License Version 2 or later (the "GPL"), * or the GNU Lesser General Public License Version 2.1 or later (the "LGPL"), * use your version of this file under the terms of the CPL, indicate your'

1.191 unboundid-ldapsdk-commercial-edition 3.2.1
1.191.1 Available under license:

Found license 'GNU Lesser General Public License' in 'All Rights Reserved. This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License (GPLv2 only) or the terms of the GNU Lesser General Public License (LGPLv2.1 only) as published by the Free Software Foundation. This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program; if not, see <http://www.gnu.org/licenses>.

INFO_RC_CONFIDENTIALITY_REQUIRED=confidentiality required'

Found license 'GNU Lesser General Public License' in '* All Rights Reserved. This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License (GPLv2 only) or the terms of the GNU Lesser General Public License (LGPLv2.1 only) as published by the Free Software Foundation. This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program; if not, see <http://www.gnu.org/licenses>.'

Found license 'GNU Lesser General Public License' in '* All Rights Reserved. This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License (GPLv2 only) or the terms of the GNU Lesser General Public License (LGPLv2.1 only) as published by the Free Software Foundation. This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program; if not, see <http://www.gnu.org/licenses>.'

Found license 'GNU Lesser General Public License' in '* All Rights Reserved. This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License (GPLv2 only) or the terms of the GNU Lesser General Public License (LGPLv2.1 only) as published by the Free Software Foundation. This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program; if not, see <http://www.gnu.org/licenses>.'
General Public License for more details. * You should have received a copy of the GNU General Public License * along with this program; if not, see <http://www.gnu.org/licenses>. * control if the requester does not have the appropriate permission to use that * // include that the authenticated user doesn't have permission to use the' Found license 'GNU Lesser General Public License' in '# All Rights Reserved. # This program is free software; you can redistribute it and/or modify # it under the terms of the GNU General Public License (GPLv2 only) # or the terms of the GNU Lesser General Public License (LGPLv2.1 only) # as published by the Free Software Foundation. # This program is distributed in the hope that it will be useful, # but WITHOUT ANY WARRANTY; without even the implied warranty of # MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the # GNU General Public License for more details. # You should have received a copy of the GNU General Public License # along with this program; if not, see <http://www.gnu.org/licenses>. * control if the requester does not have the appropriate permission to use that * // include that the authenticated user doesn't have permission to use the' Found license 'GNU Lesser General Public License' in '# All Rights Reserved. # This program is free software; you can redistribute it and/or modify # it under the terms of the GNU General Public License (GPLv2 only) # or the terms of the GNU Lesser General Public License (LGPLv2.1 only) # as published by the Free Software Foundation. # This program is distributed in the hope that it will be useful, # but WITHOUT ANY WARRANTY; without even the implied warranty of # MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the # GNU General Public License for more details. # You should have received a copy of the GNU General Public License # along with this program; if not, see <http://www.gnu.org/licenses>. * control if the requester does not have the appropriate permission to use that * // include that the authenticated user doesn't have permission to use the' Found license 'GNU Lesser General Public License' in '# All Rights Reserved. # This program is free software; you can redistribute it and/or modify # it under the terms of the GNU General Public License (GPLv2 only) # or the terms of the GNU Lesser General Public License (LGPLv2.1 only) # as published by the Free Software Foundation. # This program is distributed in the hope that it will be useful, # but WITHOUT ANY WARRANTY; without even the implied warranty of # MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the # GNU General Public License for more details. # You should have received a copy of the GNU General Public License # along with this program; if not, see <http://www.gnu.org/licenses>. * control if the requester does not have the appropriate permission to use that * // include that the authenticated user doesn't have permission to use the' Found license 'GNU Lesser General Public License' in '# All Rights Reserved. # This program is free software; you can redistribute it and/or modify # it under the terms of the GNU General Public License (GPLv2 only) # or the terms of the GNU Lesser General Public License (LGPLv2.1 only) # as published by the Free Software Foundation. # This program is distributed in the hope that it will be useful, # but WITHOUT ANY WARRANTY; without even the implied warranty of # MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the # GNU General Public License for more details. # You should have received a copy of the GNU General Public License # along with this program; if not, see <http://www.gnu.org/licenses>. * control if the requester does not have the appropriate permission to use that * // include that the authenticated user doesn't have permission to use the' Found license 'GNU Lesser General Public License' in '# All Rights Reserved. # This program is free software; you can redistribute it and/or modify # it under the terms of the GNU General Public License (GPLv2 only) # or the terms of the GNU Lesser General Public License (LGPLv2.1 only) # as published by the Free Software Foundation. # This program is distributed in the hope that it will be useful, # but WITHOUT ANY WARRANTY; without even the implied warranty of # MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the # GNU General Public License for more details. # You should have received a copy of the GNU General Public License # along with this program; if not, see <http://www.gnu.org/licenses>.
Found license 'GNU Lesser General Public License' in "* All Rights Reserved. * This program is free software; you can redistribute it and/or modify * it under the terms of the GNU General Public License (GPLv2 only) * or the terms of the GNU Lesser General Public License (LGPLv2.1 only) * as published by the Free Software Foundation. * This program is distributed in the hope that it will be useful, * but WITHOUT ANY WARRANTY; without even the implied warranty of * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the * GNU General Public License for more details. * You should have received a copy of the GNU General Public License * along with this program; if not, see <http://www.gnu.org/licenses>*. * version. * version."

Found license 'GNU Lesser General Public License' in "* All Rights Reserved. * This program is free software; you can redistribute it and/or modify * it under the terms of the GNU General Public License (GPLv2 only) * or the terms of the GNU Lesser General Public License (LGPLv2.1 only) * as published by the Free Software Foundation. * This program is distributed in the hope that it will be useful, * but WITHOUT ANY WARRANTY; without even the implied warranty of * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the * GNU General Public License for more details. * You should have received a copy of the GNU General Public License * along with this program; if not, see <http://www.gnu.org/licenses>*. * filters (which are an UnboundID-proprietary mechanism for performing advanced"

Found license 'GNU Lesser General Public License' in "* All Rights Reserved. * This program is free software; you can redistribute it and/or modify * it under the terms of the GNU General Public License (GPLv2 only) * or the terms of the GNU Lesser General Public License (LGPLv2.1 only) * as published by the Free Software Foundation. * This program is distributed in the hope that it will be useful, * but WITHOUT ANY WARRANTY; without even the implied warranty of * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the * GNU General Public License for more details. * You should have received a copy of the GNU General Public License * along with this program; if not, see <http://www.gnu.org/licenses>*. * confidentiality required

INFO_RC_CONFIDENTIALITY_REQUIRED("confidentiality required");"

Found license 'GNU Lesser General Public License' in "* All Rights Reserved. * This program is free software; you can redistribute it and/or modify * it under the terms of the GNU General Public License (GPLv2 only) * or the terms of the GNU Lesser General Public License (LGPLv2.1 only) * as published by the Free Software Foundation. * This program is distributed in the hope that it will be useful, * but WITHOUT ANY WARRANTY; without even the implied warranty of * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the * GNU General Public License for more details. * You should have received a copy of the GNU General Public License * along with this program; if not, see <http://www.gnu.org/licenses>*. * should be that of a user that has permission to use this control. The'"

Found license 'GNU Lesser General Public License' in "* All Rights Reserved. * This program is free software; you can redistribute it and/or modify * it under the terms of the GNU General Public License (GPLv2 only) * or the terms of the GNU Lesser General Public License (LGPLv2.1 only) * as published by the Free Software Foundation. * This program is distributed in the hope that it will be useful, * but WITHOUT ANY WARRANTY; without even the implied warranty of * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the * GNU General Public License for more details. * You should have received a copy of the GNU General Public License * along with this program; if not, see <http://www.gnu.org/licenses>*. * This class provides support for an UnboundID-proprietary SASL mechanism that'"

Found license 'GNU Lesser General Public License' in "* All Rights Reserved. * This program is free software; you can redistribute it and/or modify * it under the terms of the GNU General Public License (GPLv2 only) * or the terms of the GNU Lesser General Public License (LGPLv2.1 only) * as published by the Free Software Foundation. * This program is distributed in the hope that it will be useful, * but WITHOUT ANY WARRANTY; without even the implied warranty of * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the * GNU General Public License for more details. * You should have received a copy of the GNU General Public License * along with this program; if not, see <http://www.gnu.org/licenses>*. * The OID for a proprietary control that can be used to indicate that the'"
Found license 'GNU Lesser General Public License' in "* All Rights Reserved. * This program is free software; you can redistribute it and/or modify * it under the terms of the GNU General Public License (GPLv2 only) * or the terms of the GNU Lesser General Public License (LGPLv2.1 only) * as published by the Free Software Foundation. * This program is distributed in the hope that it will be useful, * but WITHOUT ANY WARRANTY; without even the implied warranty of * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the * GNU General Public License for more details. * You should have received a copy of the GNU General Public License * along with this program; if not, see <http://www.gnu.org/licenses>. * // include that the authenticated user doesn't have permission to use the'

Found license 'GNU Lesser General Public License' in "* All Rights Reserved. * This program is free software; you can redistribute it and/or modify * it under the terms of the GNU General Public License (GPLv2 only) * or the terms of the GNU Lesser General Public License (LGPLv2.1 only) * as published by the Free Software Foundation. * This program is distributed in the hope that it will be useful, * but WITHOUT ANY WARRANTY; without even the implied warranty of * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the * GNU General Public License for more details. * You should have received a copy of the GNU General Public License * along with this program; if not, see <http://www.gnu.org/licenses>. * The int value for the CONFIDENTIALITY_REQUIRED result code. public static final int CONFIDENTIALITY_REQUIRED = ResultCode.CONFIDENTIALITY_REQUIRED_INT_VALUE;"
which will permanently delete the target entry even if the server would have otherwise performed a soft delete
operation to hide the entry for a period of time before deleting it.
INFO_LDAPMODIFY_ARG_DESCRIPTION_HARD_DELETE("Indicates that all delete requests should include
the UnboundID-proprietary 'hard delete' request control, which will permanently delete the target entry even if the
server would have otherwise performed a soft delete operation to hide the entry for a period of time before deleting
it."). * Indicates that all add requests should include the UnboundID-proprietary 'ignore NO-USER-
MODIFICATION' request control to permit setting values for certain operational attributes not normally permitted
to be provided by external clients.
INFO_LDAPMODIFY_ARG_DESCRIPTION_IGNORE_NO_USER_MOD("Indicates that all add requests should
include the UnboundID-proprietary 'ignore NO-USER-MODIFICATION' request control to permit setting values
for certain operational attributes not normally permitted to be provided by external clients."). * Include the
UnboundID-proprietary 'name with entryUUID' request control in all add requests sent to the server to indicate that
the server should use the entryUUID operational attribute as the naming attribute for the resulting entry instead of
the provided RDN. INFO_LDAPMODIFY_ARG_DESCRIPTION_NAME_WITH_ENTRY_UUID("Include the
UnboundID-proprietary 'name with entryUUID' request control in all add requests sent to the server to indicate that
the server should use the entryUUID operational attribute as the naming attribute for the resulting entry instead of
the provided RDN."). * Indicates that all operations should include the UnboundID-proprietary 'operation purpose'
request control to provide the specified reason for the operation.
INFO_LDAPMODIFY_ARG_DESCRIPTION_OPERATION_PURPOSE("Indicates that all operations should
include the UnboundID-proprietary 'operation purpose' request control to provide the specified reason for the
operation."). * Indicates that all add and modify requests that target either the "{0}" or "{1}" attribute should include
the UnboundID-proprietary 'password validation details' request control to indicate that the response should include
information about the password quality requirements that the server will impose for the target user's password and
whether the provided password satisfies each of those constraints.
INFO_LDAPMODIFY_ARG_DESCRIPTION_PASSWORD_VALIDATION_DETAILS("Indicates that all add
and modify requests that target either the "{0}" or "{1}" attribute should include the UnboundID-proprietary
'password validation details' request control to indicate that the response should include information about the
password quality requirements that the server will impose for the target user's password and whether the provided
password satisfies each of those constraints."). * Indicates that any modify operation that targets either the "{0}" or
"{1}" attribute should include the UnboundID-proprietary 'purge current password' request control. This will
indicate that the server should purge the current password from the entry (even if it would have otherwise
been retired for a brief period of time).
INFO_LDAPMODIFY_ARG_DESCRIPTION_PURGE_CURRENT_PASSWORD("Indicates that any modify
operation that targets either the "{0}" or "{1}" attribute should include the UnboundID-proprietary 'purge current
password' request control. This will indicate that the server should purge the current password from the entry
(even if it would have otherwise been retired for a brief period of time."). * Indicates that all operation requests
should include the UnboundID-proprietary 'replication repair' request control. This will cause the change to be
applied only to the target directory server instance but not to any other server in the replication topology.
INFO_LDAPMODIFY_ARG_DESCRIPTION_REPLICATION_REPAIR("Indicates that all operation requests
should include the UnboundID-proprietary 'replication repair' request control. This will cause the change to be
applied only to the target directory server instance but not to any other server in the replication topology."). * Indicates that any modify operation that targets either the "{0}" or "{1}" attribute should include the UnboundID-
proprietary 'retire current password' request control. This will indicate that the server should continue to allow the
user to authenticate with their current password (in addition to the new password) for a brief period of time.
INFO_LDAPMODIFY_ARG_DESCRIPTION_RETIRE_CURRENT_PASSWORD("Indicates that any modify
operation that targets either the "{0}" or "{1}" attribute should include the UnboundID-proprietary 'retire current
password' request control. This will indicate that the server should continue to allow the user to authenticate with
their current password (in addition to the new password) for a brief period of time."). * Indicates that all delete
requests should include the UnboundID-proprietary 'soft delete' request control, which indicates that the server should hide the entry for a period of time before deleting it so that it may be restored with an undelete operation if the delete should be reverted. INFO_LDAPMODIFY_ARG_DESCRIPTION_SOFT_DELETE("Indicates that all delete requests should include the UnboundID-proprietary 'soft delete' request control, which indicates that the server should hide the entry for a period of time before deleting it so that it may be restored with an undelete operation if the delete should be reverted."). * Indicates that all operations should include the UnboundID-proprietary 'suppress operational attribute updates' request control to indicate that the server should not apply any updates to the specified operational attributes. The value may be one of 'last-access-time', 'last-login-time', 'last-login-ip', or 'lastmod'.

INFO_LDAPMODIFY_ARG_DESCRIPTION_SUPPRESS_OP_ATTR_UPDATES("Indicates that all operations should include the UnboundID-proprietary 'suppress operational attribute updates' request control to indicate that the server should not apply any updates to the specified operational attributes. The value may be one of 'last-access-time', 'last-login-time', 'last-login-ip', or 'lastmod'."), * Indicates that the tool should include the UnboundID-proprietary 'suppress referential integrity updates' request control in all delete and modify DN operations to indicate that the server should not perform any referential integrity processing for those operations.

INFO_LDAPMODIFY_ARG_DESCRIPTION_SUPPRESS_REFERINT_UPDATES("Indicates that the tool should include the UnboundID-proprietary 'suppress referential integrity updates' request control in all delete and modify DN operations to indicate that the server should not perform any referential integrity processing for those operations."). * Indicates that all search requests should include the UnboundID-proprietary account usable request control to request that each search result entry returned include a response control with information about the password policy usability state for the entry.

INFO_LDAPSEARCH_ARG_DESCRIPTION_ACCOUNT_USABLE("Indicates that all search requests should include the UnboundID-proprietary account usable request control to request that each search result entry returned include a response control with information about the password policy usability state for the entry."). * Indicates that all search requests should include the UnboundID-proprietary 'exclude branch' request control to indicate that matching entries below the specified base DN should be excluded from search results. This argument may be provided multiple times if multiple branches should be excluded.

INFO_LDAPSEARCH_ARG_DESCRIPTION_EXCLUDE_BRANCH("Indicates that all search requests should include the UnboundID-proprietary 'exclude branch' request control to indicate that matching entries below the specified base DN should be excluded from search results. This argument may be provided multiple times if multiple branches should be excluded."). * Indicates that all bind requests should include the UnboundID-proprietary 'get authorization entry' request control to request that the server return the specified attribute (or collection of attributes, in the case of a special identifier like '*' to indicate all user attributes or '+' to indicate all operational attributes) from the authenticated user's entry. This argument may be provided multiple times to specify multiple attributes to request.

INFO_LDAPSEARCH_ARG_DESCRIPTION_GET_AUTHZ_ENTRY_ATTR("Indicates that all bind requests should include the UnboundID-proprietary 'get authorization entry' request control to request that the server return the specified attribute (or collection of attributes, in the case of a special identifier like '*' to indicate all user attributes or '+' to indicate all operational attributes) from the authenticated user's entry. This argument may be provided multiple times to specify multiple attributes to request."). * Indicates that all search requests should include the UnboundID-proprietary 'get effective rights' request control to return information about the access control rights that a user has when interacting with each matching entry. This argument may be provided multiple times to specify multiple attributes.

INFO_LDAPSEARCH_ARG_DESCRIPTION_GET_EFFECTIVE_RIGHTS_ATTR("Indicates that all search requests should include the UnboundID-proprietary 'get effective rights' request control to return information about the access control rights that a user has when interacting with each matching entry. This argument may be provided multiple times to specify multiple attributes."). * Indicates that all search requests should include the UnboundID-proprietary 'get effective rights' request control to return information about the access control rights that a user has when interacting with each matching entry.
INFO_LDAPSEARCH_ARG_DESCRIPTION_GET_EFFECTIVE_RIGHTS_AUTHZID("Indicates that all search requests should include the UnboundID-proprietary 'get effective rights' request control to return information about the access control rights the specified user has when interacting with each matching entry."). * Indicates that all bind requests should include the UnboundID-proprietary 'get user resource limits' request control to request that the server return information about resource limits (e.g., size limit, time limit, idle time limit, etc.) imposed for the user.

INFO_LDAPSEARCH_ARG_DESCRIPTION_GET_USER_RESOURCE_LIMITS("Indicates that all bind requests should include the UnboundID-proprietary 'get user resource limits' request control to request that the server return information about resource limits (e.g., size limit, time limit, idle time limit, etc.) imposed for the user."). * Indicates that all search requests should include the UnboundID-proprietary 'return conflict entries request control' to indicate that the server may return any replication conflict entries that match the search criteria. Replication conflict entries are normally excluded from search results.

INFO_LDAPSEARCH_ARG_DESCRIPTION_INCLUDE_REPL_CONFLICTS("Indicates that all search requests should include the UnboundID-proprietary 'return conflict entries request control' to indicate that the server may return any replication conflict entries that match the search criteria.

INFO_LDAPSEARCH_ARG_DESCRIPTION_INCLUDE_SOFT_DELETED("Indicates that all search requests should include the UnboundID-proprietary 'soft-deleted entry access' request control to indicate that the server may return any soft-deleted entries that match the search criteria. Soft-deleted entries are normally excluded from search results. The value for this argument must be one of: 'with-non-deleted-entries' (indicates that both regular and soft-deleted entries should be returned), 'without-non-deleted-entries' (indicates that only soft-deleted entries should be returned), or 'deleted-entries-in-undeleted-form' (returns only soft-deleted entries in the form in the form the entry had before it was deleted).

INFO_LDAPSEARCH_ARG_DESCRIPTION_MATCHING_ENTRY_COUNT_CONTROL("Indicates that all search requests should include the UnboundID-proprietary 'matching entry count' request control, which indicates that the server should return information about the number of entries that match the search criteria. The maximum number of entries to examine must be specified, which helps indicate whether an exact count or an estimate will be returned. If alwaysExamine is specified and the number of candidates is less than the examine count, then each candidate will be examined to verify that it matches the criteria and would actually be returned to the client in a search. If allowUnindexed is specified, then the count will be allowed to be processed even if the search is unindexed (and may take a very long time to complete). If debug is specified, then additional debug information may be included in the output.

INFO_LDAPSEARCH_ARG_DESCRIPTION_OPERATION_PURPOSE("Indicates that all search requests should include the UnboundID-proprietary 'operation purpose' request control to provide the specified reason for the operation.

INFO_LDAPSEARCH_ARG_DESCRIPTION_REAL_ATTRIBUTES_ONLY("Indicates that all search requests should include the UnboundID-proprietary 'real attributes only' request control to indicate that the server should not include any virtual attributes in entries that are returned.

Open Source Used In security_automation identity-guest-cs 2082
INFO_LDAPSEARCH_ARG_DESCRIPTION_REAL_ATTRS_ONLY("Indicates that all search requests should include the UnboundID-proprietary 'real attributes only' request control to indicate that the server should not include any virtual attributes in entries that are returned."). * Indicates that all operations should include the UnboundID-proprietary 'suppress operational attribute updates' request control to indicate that the server should not apply any updates to the specified operational attributes. The value may be one of 'last-access-time', 'last-login-time', 'last-login-ip', or 'lastmod'.

INFO_LDAPSEARCH_ARG_DESCRIPTION_SUPPRESS_OP_ATTR_UPDATES("Indicates that all operations should include the UnboundID-proprietary 'suppress operational attribute updates' request control to indicate that the server should not apply any updates to the specified operational attributes. The value may be one of 'last-access-time', 'last-login-time', 'last-login-ip', or 'lastmod'."). * Indicates that all search requests should include the UnboundID-proprietary 'virtual attributes only' request control to indicate that the server should only include virtual attributes in entries that are returned.

INFO_LDAPSEARCH_ARG_DESCRIPTION_VIRTUAL_ATTRS_ONLY("Indicates that all search requests should include the UnboundID-proprietary 'virtual attributes only' request control to indicate that the server should only include virtual attributes in entries that are returned.");''

Found license 'GNU Lesser General Public License' in '* All Rights Reserved. * This program is free software; you can redistribute it and/or modify * it under the terms of the GNU General Public License (GPLv2 only) * or the terms of the GNU Lesser General Public License (LGPLv2.1 only) * as published by the Free Software Foundation. * This program is distributed in the hope that it will be useful, * but WITHOUT ANY WARRANTY; without even the implied warranty of * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the * GNU General Public License for more details. * You should have received a copy of the GNU General Public License * along with this program; if not, see <http://www.gnu.org/licenses>. "requires that the authenticated user have permission to use the " +

Found license 'GNU Lesser General Public License' in '* All Rights Reserved. * This program is free software; you can redistribute it and/or modify * it under the terms of the GNU General Public License (GPLv2 only) * or the terms of the GNU Lesser General Public License (LGPLv2.1 only) * as published by the Free Software Foundation. * This program is distributed in the hope that it will be useful, * but WITHOUT ANY WARRANTY; without even the implied warranty of * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the * GNU General Public License for more details. * You should have received a copy of the GNU General Public License * along with this program; if not, see <http://www.gnu.org/licenses>. * The quality of protection that should be used for any communication that occurs after the authentication has completed. Allowed values are 'auth' (for just authentication with no communication protection), 'auth-int' (for integrity protection for communication, which does not encrypt but ensures that the communication cannot be imperceptibly altered by a man in the middle attack), and 'auth-conf' (for confidentiality protection for communication, which encrypts the communication so that it cannot be deciphered by a third-party observer). If no value is specified, then a default of 'auth' will be assumed. If any of multiple qualities of protection will be considered acceptable then the permissible QoP values may be separated by commas and listed in order from most desirable to least desirable.

INFO_SASL_DIGEST_MD5_OPTION_QOP("The quality of protection that should be used for any communication that occurs after the authentication has completed. Allowed values are 'auth' (for just authentication with no communication protection), 'auth-int' (for integrity protection for communication, which does not encrypt but ensures that the communication cannot be imperceptibly altered by a man in the middle attack), and 'auth-conf' (for confidentiality protection for communication, which encrypts the communication so that it cannot be deciphered by a third-party observer). If no value is specified, then a default of 'auth' will be assumed. If any of multiple qualities of protection will be considered acceptable then the permissible QoP values may be separated by commas and listed in order from most desirable to least desirable."). * The quality of protection that should be used for any communication that occurs after the authentication has completed. Allowed values are 'auth' (for just authentication with no communication protection), 'auth-int' (for integrity protection for communication, which does not encrypt but ensures that the communication cannot be imperceptibly altered by a man in the middle attack), and 'auth-conf'
(for confidentiality protection for communication, which encrypts the communication so that it cannot be deciphered by a third-party observer). If no value is specified, then a default of 'auth' will be assumed. If any of multiple qualities of protection will be considered acceptable then the permissible QoP values may be separated by commas and listed in order from most desirable to least desirable. INFO_SASL_GSSAPI_OPTION_QOP("The quality of protection that should be used for any communication that occurs after the authentication has completed. Allowed values are 'auth' (for just authentication with no communication protection), 'auth-int' (for integrity protection for communication, which does not encrypt but ensures that the communication cannot be imperceptibly altered by a man in the middle attack), and 'auth-conf' (for confidentiality protection for communication, which encrypts the communication so that it cannot be deciphered by a third-party observer). If no value is specified, then a default of 'auth' will be assumed. If any of multiple qualities of protection will be considered acceptable then the permissible QoP values may be separated by commas and listed in order from most desirable to least desirable.").'

Found license 'GNU Lesser General Public License' in '/* All Rights Reserved. * This program is free software; you can redistribute it and/or modify * it under the terms of the GNU General Public License (GPLv2 only) * or the terms of the GNU Lesser General Public License (LGPLv2.1 only) * as published by the Free Software Foundation. * This program is distributed in the hope that it will be useful, * but WITHOUT ANY WARRANTY; without even the implied warranty of * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the * GNU General Public License for more details. * You should have received a copy of the GNU General Public License * along with this program; if not, see <http://www.gnu.org/licenses>. * form of SASL integrity or confidentiality, and should not be used for other'

Found license 'GNU Lesser General Public License' in '# All Rights Reserved. # This program is free software; you can redistribute it and/or modify # it under the terms of the GNU General Public License (GPLv2 only) # or the terms of the GNU Lesser General Public License (LGPLv2.1 only) # as published by the Free Software Foundation. # This program is distributed in the hope that it will be useful, # but WITHOUT ANY WARRANTY; without even the implied warranty of # MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the # GNU General Public License for more details. # You should have received a copy of the GNU General Public License # along with this program; if not, see <http://www.gnu.org/licenses>.'

1.192 retrofit 2.9.0

1.192.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2011 Square, Inc.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
*/
* Copyright (C) 2018 Square, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1083957721_1598378129.06/0/retrofit-2-9-0-sources-jar/retrofit2/Retrofit.kt
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2013 Square, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *      http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1083957721_1598378129.06/0/retrofit-2-9-0-sources-jar/retrofit2/Invocation.java
* /opt/ws_local/PERMITS_SQL/1083957721_1598378129.06/0/retrofit-2-9-0-sources-jar/retrofit2/internal/EverythingIsNonNull.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2015 Square, Inc.
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
*    http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1083957721_1598378129.06/0/retrofit-2-9-0-sources-jar/retrofit2/http/Url.java
* /opt/ws_local/PERMITS_SQL/1083957721_1598378129.06/0/retrofit-2-9-0-sources-jar/retrofit2/OkHttpCall.java
* /opt/ws_local/PERMITS_SQL/1083957721_1598378129.06/0/retrofit-2-9-0-sources-jar/retrofit2/BuiltInConverters.java
* /opt/ws_local/PERMITS_SQL/1083957721_1598378129.06/0/retrofit-2-9-0-sources-jar/retrofit2/Response.java
* /opt/ws_local/PERMITS_SQL/1083957721_1598378129.06/0/retrofit-2-9-0-sources-jar/retrofit2/http/OPTIONS.java
* /opt/ws_local/PERMITS_SQL/1083957721_1598378129.06/0/retrofit-2-9-0-sources-jar/retrofit2/ServiceMethod.java
* /opt/ws_local/PERMITS_SQL/1083957721_1598378129.06/0/retrofit-2-9-0-sources-jar/retrofit2/RequestFactory.java
* /opt/ws_local/PERMITS_SQL/1083957721_1598378129.06/0/retrofit-2-9-0-sources-jar/retrofit2/CallAdapter.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2019 Square, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
*    http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1083957721_1598378129.06/0/retrofit-2-9-0-sources-jar/retrofit2/SkipCallbackExecutorImpl.java
* /opt/ws_local/PERMITS_SQL/1083957721_1598378129.06/0/retrofit-2-9-0-sources-jar/retrofit2/SkipCallbackExecutor.java
* /opt/ws_local/PERMITS_SQL/1083957721_1598378129.06/0/retrofit-2-9-0-sources-jar/retrofit2/http/Tag.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2017 Square, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1083957721_1598378129.06/0/retrofit-2-9-0-sources-jar/retrofit2/OptionalConverterFactory.java
No license file was found, but licenses were detected in source scan.

/*/ 
 * Copyright (C) 2016 Square, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1083957721_1598378129.06/0/retrofit-2-9-0-sources-jar/retrofit2/CompletableFutureCallAdapterFactory.java
* /opt/ws_local/PERMITS_SQL/1083957721_1598378129.06/0/retrofit-2-9-0-sources-jar/retrofit2/HeaderMap.java
* /opt/ws_local/PERMITS_SQL/1083957721_1598378129.06/0/retrofit-2-9-0-sources-jar/retrofit2/HttpException.java

No license file was found, but licenses were detected in source scan.

*/

* Copyright (C) 2014 Square, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* *
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1083957721_1598378129.06/0/retrofit-2-9-0-sources-jar/retrofit2/FieldMap.java
* /opt/ws_local/PERMITS_SQL/1083957721_1598378129.06/0/retrofit-2-9-0-sources-jar/retrofit2/HttpStream.java
* /opt/ws_local/PERMITS_SQL/1083957721_1598378129.06/0/retrofit-2-9-0-sources-jar/retrofit2/HttpMap.java
* /opt/ws_local/PERMITS_SQL/1083957721_1598378129.06/0/retrofit-2-9-0-sources-jar/retrofit2/FieldMap.java
* /opt/ws_local/PERMITS_SQL/1083957721_1598378129.06/0/retrofit-2-9-0-sources-jar/retrofit2/HttpResponse.java

No license file was found, but licenses were detected in source scan.

*/

* Copyright (C) 2008 Google Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* *
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1083957721_1598378129.06/0/retrofit-2-9-0-sources-jar/retrofit2/Utils.java

1.193 snake-yaml 1.27
1.193.1 Available under license :
Found license 'GNU Lesser General Public License' in '// This module is multi-licensed and may be used under the
terms // EPL, Eclipse Public License, V1.0 or later, http://www.eclipse.org/legal // LGPL, GNU Lesser General
Public License, V2.1 or later, http://www.gnu.org/licenses/lgpl.html // GPL, GNU General Public License, V2 or
/ BSD.'

Found license 'Eclipse Public License 1.0' in '// This module is multi-licensed and may be used under the terms //
EPL, Eclipse Public License, V1.0 or later, http://www.eclipse.org/legal // LGPL, GNU Lesser General Public
License, V2.1 or later, http://www.gnu.org/licenses/lgpl.html // GPL, GNU General Public License, V2 or later,
http://www.gnu.org/licenses/gpl.html // AL, Apache License, V2.0 or later, http://www.apache.org/licenses //
/ BSD.'

1.194 asm 7.1
1.194.1 Available under license :
No license file was found, but licenses were detected in source scan.

// All rights reserved.
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions
// are met:
// 1. Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
// 2. Redistributions in binary form must reproduce the above copyright
// notice, this list of conditions and the following disclaimer in the
// documentation and/or other materials provided with the distribution.
// 3. Neither the name of the copyright holders nor the names of its
// this software without specific prior written permission.

Found in path(s):
* /opt/cola/permits/1135892428_1613626883.3/0/asm-7-1-sources-6-jar/org/objectweb/asm/ClassWriter.java
* /opt/cola/permits/1135892428_1613626883.3/0/asm-7-1-sources-6-jar/org/objectweb/asm/Handler.java
* /opt/cola/permits/1135892428_1613626883.3/0/asm-7-1-sources-6-jar/org/objectweb/asm/Frame.java
No license file was found, but licenses were detected in source scan.

2011 INRIA, France Telecom

* All rights reserved.

* Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* 3. Neither the name of the copyright holders nor the names of its
* contributors may be used to endorse or promote products derived from
* this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
* LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
* CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
* SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
* INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
* CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
* THE POSSIBILITY OF SUCH DAMAGE.

Found in path(s):
* /opt/cola/permits/1135892428_1613626883.3/0/asm-7-1-sources-6-jar/org/objectweb/asm/package.html
* /opt/cola/permits/1135892428_1613626883.3/0/asm-7-1-sources-6-jar/org/objectweb/asm/signature/package.html

1.195 jandex 2.2.3.Final
1.195.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.196 commons-io 2.11.0

1.196.1 Available under license:
Apache Commons IO
Copyright 2002-2021 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (https://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and
(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0
Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.197 spring-security-crypto 5.4.9

1.197.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2011-2016 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License);
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1239873012_1639093473.79/0/spring-security-crypto-5-4-9-sources-jar/org/springframework/security/crypto/keygen/KeyGenerators.java
* /opt/cola/permits/1239873012_1639093473.79/0/spring-security-crypto-5-4-9-sources-jar/org/springframework/security/crypto/keygen/HexEncodingStringKeyGenerator.java
* /opt/cola/permits/1239873012_1639093473.79/0/spring-security-crypto-5-4-9-sources-jar/org/springframework/security/crypto/keygen/SecureRandomBytesKeyGenerator.java
* /opt/cola/permits/1239873012_1639093473.79/0/spring-security-crypto-5-4-9-sources-jar/org/springframework/security/crypto/encrypt/HexEncodingTextEncryptor.java
* /opt/cola/permits/1239873012_1639093473.79/0/spring-security-crypto-5-4-9-sources-jar/org/springframework/security/crypto/keygen/SharedKeyGenerator.java
* /opt/cola/permits/1239873012_1639093473.79/0/spring-security-crypto-5-4-9-sources-jar/org/springframework/security/crypto/encrypt/BouncyCastleAesGcmBytesEncryptor.java
* /opt/cola/permits/1239873012_1639093473.79/0/spring-security-crypto-5-4-9-sources-jar/org/springframework/security/crypto/keygen/StringKeyGenerator.java
No license file was found, but licenses were detected in source scan.

// Permission to use, copy, modify, and distribute this software for any
// purpose with or without fee is hereby granted, provided that the above
// copyright notice and this permission notice appear in all copies.

Found in path(s):
* /opt/cola/permits/1239873012_1639093473.79/0/spring-security-crypto-5-4-9-sources-jar/org/springframework/security/crypto/bcrypt/BCrypt.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2016 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *      https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

/**
 * Base64 encoder which is a reduced version of Robert Harder's public domain
 * implementation (version 2.3.7). See <a href="http://iharder.sourceforge.net/current/java/base64/">http://iharder.sourceforge.net/current/java/base64/</a>
 */
* for more information.
* <p>
* For internal use only.
*
* @author Luke Taylor
* @since 3.0
* @deprecated Use java.util.Base64
*/

Found in path(s):
* /opt/cola/permits/1239873012_1639093473.79/0/spring-security-crypto-5-4-9-sources-jar/org/springframework/security/crypto/codec/Base64.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2002-2019 the original author or authors.
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
*   https://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1239873012_1639093473.79/0/spring-security-crypto-5-4-9-sources-jar/org/springframework/security/crypto/argon2/Argon2EncodingUtils.java
* /opt/cola/permits/1239873012_1639093473.79/0/spring-security-crypto-5-4-9-sources-jar/org/springframework/security/crypto/scrypt/SCryptPasswordEncoder.java
* /opt/cola/permits/1239873012_1639093473.79/0/spring-security-crypto-5-4-9-sources-jar/org/springframework/security/crypto/argon2/Argon2PasswordEncoder.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2011-2018 the original author or authors.
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
*   https://www.apache.org/licenses/LICENSE-2.0
* 
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1239873012_1639093473.79/0/spring-security-crypto-5-4-9-sources-jar/org/springframework/security/crypto/password/Digester.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2016 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *      https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1239873012_1639093473.79/0/spring-security-crypto-5-4-9-sources-jar/org/springframework/security/crypto/codec/Hex.java
* /opt/cola/permits/1239873012_1639093473.79/0/spring-security-crypto-5-4-9-sources-jar/org/springframework/security/crypto/codec/Utf8.java
* /opt/cola/permits/1239873012_1639093473.79/0/spring-security-crypto-5-4-9-sources-jar/org/springframework/security/crypto/codec/package-info.java
* /opt/cola/permits/1239873012_1639093473.79/0/spring-security-crypto-5-4-9-sources-jar/org/springframework/security/crypto/password/AbstractPasswordEncoder.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2011-2020 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *      https://www.apache.org/licenses/LICENSE-2.0
 *
*/
* Unless required by applicable law or agreed to in writing, software
  * distributed under the License is distributed on an "AS IS" BASIS,
  * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
  * See the License for the specific language governing permissions and
  * limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1239873012_1639093473.79/0/spring-security-crypto-5-4-9-sources-jar/org/springframework/security/crypto/encrypt/Encryptors.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2017 the original author or authors.
 * 
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 *    https://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1239873012_1639093473.79/0/spring-security-crypto-5-4-9-sources-jar/org/springframework/security/crypto/factory/PasswordEncoderFactories.java
* /opt/cola/permits/1239873012_1639093473.79/0/spring-security-crypto-5-4-9-sources-jar/org/springframework/security/crypto/password/Md4.java
* /opt/cola/permits/1239873012_1639093473.79/0/spring-security-crypto-5-4-9-sources-jar/org/springframework/security/crypto/password/PasswordEncoderUtils.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2018 the original author or authors.
 * 
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 *    https://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
  * /opt/cola/permits/1239873012_1639093473.79/0/spring-security-crypto-5-4-9-sources-jar/org/springframework/security/crypto/password/MessageDigestPasswordEncoder.java
  * /opt/cola/permits/1239873012_1639093473.79/0/spring-security-crypto-5-4-9-sources-jar/org/springframework/security/crypto/password/Md4PasswordEncoder.java
  * /opt/cola/permits/1239873012_1639093473.79/0/spring-security-crypto-5-4-9-sources-jar/org/springframework/security/crypto/keygen/Base64StringKeyGenerator.java
  * /opt/cola/permits/1239873012_1639093473.79/0/spring-security-crypto-5-4-9-sources-jar/org/springframework/security/crypto/password/Pbkdf2PasswordEncoder.java
  * /opt/cola/permits/1239873012_1639093473.79/0/spring-security-crypto-5-4-9-sources-jar/org/springframework/security/crypto/password/DelegatingPasswordEncoder.java
  * /opt/cola/permits/1239873012_1639093473.79/0/spring-security-crypto-5-4-9-sources-jar/org/springframework/security/crypto/password/LdapShaPasswordEncoder.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2011 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
  * /opt/cola/permits/1239873012_1639093473.79/0/spring-security-crypto-5-4-9-sources-jar/org/springframework/security/crypto/bcrypt/BCryptPasswordEncoder.java

1.198 listenablefuture 9999.0-empty-to-avoid-conflict-with-guava
1.198.1 Available under license:

Apache License

Version 2.0, January 2004

http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."
"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

You must give any other recipients of the Work or Derivative Works a copy of this License; and
You must cause any modified files to carry prominent notices stating that You changed the files; and
You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and
reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

1.199 spring-amqp 2.3.11

1.199.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015-2019 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *      https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
Found in path(s):
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-jar/org/springframework/amqp/event/AmqpEvent.java
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-jar/org/springframework/amqp/support/converter/ContentTypeDelegatingMessageConverter.java
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-jar/org/springframework/amqp/support/ConsumerTagStrategy.java
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-jar/org/springframework/amqp/AmqpTimeoutException.java
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-jar/org/springframework/amqp/support/ConditionalExceptionLogger.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015-2020 the original author or authors.
 * 
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 *     https://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/ cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-jar/org/springframework/amqp/core/AmqpMessageReturnedException.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2019 the original author or authors.
 * 
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 *     https://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
*/

Found in path(s):
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-jar/org/springframework/amqp/core/Declarable.java
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-jar/org/springframework/amqp/support/converter/DefaultClassMapper.java
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-jar/org/springframework/amqp/remoting/client/AmqpClientInterceptor.java
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-jar/org/springframework/amqp/AmqpException.java
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-jar/org/springframework/amqp/support/converter/ClassMapper.java
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-jar/org/springframework/amqp/remoting/client/AmqpProxyFactoryBean.java
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-jar/org/springframework/amqp/core/CustomExchange.java
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-jar/org/springframework/amqp/support/converter/MarshallingMessageConverter.java
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-jar/org/springframework/amqp/core/Address.java
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-jar/org/springframework/amqp/core/SimpleAmqpHeaderMapper.java
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-jar/org/springframework/amqp/core/AmqpIOException.java
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-jar/org/springframework/amqp/core/BindingBuilder.java
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-jar/org/springframework/amqp/core/AmqpAdmin.java
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-jar/org/springframework/amqp/core/Exchange.java
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-jar/org/springframework/amqp/AmqpRejectAndDontRequeueException.java
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-jar/org/springframework/amqp/support/converter/AbstractJavaTypeMapper.java
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-jar/org/springframework/amqp/support/converter/SimpleMessageConverter.java
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-jar/org/springframework/amqp/core/HeadersExchange.java

/* Copyright 2018-2020 the original author or authors.

* Licensed under the Apache License, Version 2.0 (the "License);
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*/

No license file was found, but licenses were detected in source scan.
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-jar/org/springframework/amqp/support/converter/AbstractJackson2MessageConverter.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2002-2021 the original author or authors.
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-jar/org/springframework/amqp/core/Queue.java
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-jar/org/springframework/amqp/core/Message.java
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-jar/org/springframework/amqp/core/AnonymousQueue.java
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-jar/org/springframework/amqp/core/MessagePostProcessor.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2014-2019 the original author or authors.
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
* /

Found in path(s):
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-jar/org/springframework/amqp/core/MessageBuilder.java
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-jar/org/springframework/amqp/support/postprocessor/MessagePostProcessorUtils.java
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-jar/org/springframework/amqp/support/postprocessor/UnzipPostProcessor.java
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-jar/org/springframework/amqp/support/postprocessor/ZipPostProcessor.java
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-jar/org/springframework/amqp/support/postprocessor/GUnzipPostProcessor.java
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-jar/org/springframework/amqp/core/MessagePropertiesBuilder.java
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-jar/org/springframework/amqp/core/ReceiveAndReplyCallback.java
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-jar/org/springframework/amqp/support/postprocessor/AbstractDeflaterPostProcessor.java
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-jar/org/springframework/amqp/support/AmqpMessageHeaderAccessor.java
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-jar/org/springframework/amqp/core/MessageBuilderSupport.java
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-jar/org/springframework/amqp/support/postprocessor/KGZipPostProcessor.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2020 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
Found in path(s):
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-jar/org/springframework/amqp/core/ReturnedMessage.java
No license file was found, but licenses were detected in source scan.

*/

* Copyright 2016-2021 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* *
* https://www.apache.org/licenses/LICENSE-2.0
* *
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-jar/org/springframework/amqp/core/QueueBuilder.java
No license file was found, but licenses were detected in source scan.

*/

* Copyright 2014-2020 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* *
* https://www.apache.org/licenses/LICENSE-2.0
* *
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-jar/org/springframework/amqp/support/converter/MessagingMessageConverter.java
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-
/*
 * Copyright 2006-2019 the original author or authors.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * https://www.apache.org/licenses/LICENSE-2.0
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-
jar/org/springframework/amqp/utils/SerializationUtils.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2017-2019 the original author or authors.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * https://www.apache.org/licenses/LICENSE-2.0
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-
jar/org/springframework/amqp/core/Correlation.java
No license file was found, but licenses were detected in source scan.

/
* Copyright 2018-2019 the original author or authors.
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-jar/org/springframework/amqp/core/NamingStrategy.java
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-jar/org/springframework/amqp/core/UUIDNamingStrategy.java
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-jar/org/springframework/amqp/core/Declarables.java
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-jar/org/springframework/amqp/support/SendRetryContextAccessor.java
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-jar/org/springframework/amqp/support/AmqpResourceNotAvailableException.java
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-jar/org/springframework/amqp/support/converter/Jackson2XmlMessageConverter.java
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-jar/org/springframework/amqp/support/converter/Base64UrlNamingStrategy.java

No license file was found, but licenses were detected in source scan.

*/

* Copyright 2002-2020 the original author or authors.
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
Found in path(s):
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-jar/org/springframework/amqp/support/AmqpHeaders.java
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-jar/org/springframework/amqp/support/converter/DefaultJackson2JavaTypeMapper.java

No license file was found, but licenses were detected in source scan.

/**
 * Copyright 2014-2021 the original author or authors.
 * 
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 *     https://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-jar/springframework/amqp/support/postprocessor/DelegatingDecompressingPostProcessor.java

No license file was found, but licenses were detected in source scan.

/**
 * Copyright 2002-2019 the original author or authors.
 * 
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 *     https://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

/**
 * Special exception for listener implementations that want to signal that the current
 * batch of messages should be acknowledged immediately (i.e. as soon as possible) without
* rollback, and without consuming any more messages within the current transaction.
*  
* @author Dave Syer
* @author Gary Russell
*  
*/

Found in path(s):
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-jar/org/springframework/amqp/ImmediateAcknowledgeAmqpException.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2013-2019 the original author or authors.
*  
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*  
*     https://www.apache.org/licenses/LICENSE-2.0
*  
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-jar/org/springframework/amqp/core/ReplyToAddressCallback.java
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-jar/org/springframework/amqp/AmqpAuthenticationException.java
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-jar/org/springframework/amqp/utils/test/TestUtils.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2016-2019 the original author or authors.
*  
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*  
*     https://www.apache.org/licenses/LICENSE-2.0
*  

* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-
jar/org/springframework/amqp/support/converter/RemoteInvocationAwareMessageConverterAdapter.java
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-
jar/org/springframework/amqp/AmqpRemoteException.java
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-
jar/org/springframework/amqp/support/converter/SmartMessageConverter.java
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-
jar/org/springframework/amqp/core/AbstractBuilder.java
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-
jar/org/springframework/amqp/core/AsyncAmqpTemplate.java
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-
jar/org/springframework/amqp/core/AmqpApplicationContextClosedException.java
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-
jar/org/springframework/amqp/core/AmqpReplyTimeoutException.java
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-
jar/org/springframework/amqp/support/converter/AllowedListDeserializingMessageConverter.java
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-
jar/org/springframework/amqp/utils/MapBuilder.java
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-
jar/org/springframework/amqp/core/ExchangeBuilder.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2019 the original author or authors.
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
*    https://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178442_1635319266.83/0/spring-amqp-2-3-11-sources-
1.200 java-statsd-client 3.1.0

1.200.1 Available under license :

java-statsd-client. Copyright (C) 2014 youDevise, Ltd.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.201 httpasyncclient 4.1.4

1.201.1 Available under license :

Apache HttpAsyncClient
Copyright 2010-2018 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.
"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[ ]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
You may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.202 commons-digester 1.8.1
1.202.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.
5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Apache Commons Digester
Copyright 2001-2008 The Apache Software Foundation

This product includes software developed by
The Apache Software Foundation (http://www.apache.org/).

1.203 aalto-xml 1.0.0

1.203.1 Available under license:

This copy of Aalto XML processor is licensed under the
Apache (Software) License, version 2.0 ("the License").
See the License for details about distribution rights, and the
specific rights regarding derivate works.

You may obtain a copy of the License at:

    http://www.apache.org/licenses/

A copy is also included with both the the downloadable source code package
and jar that contains class bytecodes, as file "ASL 2.0". In both cases,
that file should be located next to this file: in source distribution
the location should be "release-notes/asl"; and in jar "META-INF/"
Apache Avro Guava Dependencies
Copyright 2009-2017 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICExx text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[[]]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

License for Guava classes included in this binary artifact:

Copyright: 2006-2015 The Guava Authors
License: http://www.apache.org/licenses/LICENSE-2.0 (see above)

1.205 javax-inject 1
1.205.1 Available under license:
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensee" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.
2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "["]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

/*
 * Copyright (C) 2009 The JSR-330 Expert Group
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * */
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

1.206 abego-treelayout-core 1.0.1

1.206.1 Available under license:

[The "BSD license"]
Copyright (c) 2011, abego Software GmbH, Germany (http://www.abego.org)
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice,
   this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice,
   this list of conditions and the following disclaimer in the documentation
   and/or other materials provided with the distribution.
3. Neither the name of the abego Software GmbH nor the names of its
   contributors may be used to endorse or promote products derived from this
   software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE
LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE
POSSIBILITY OF SUCH DAMAGE.

1.207 jcl-over-slf4j 1.7.32

1.207.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2001-2004 The Apache Software Foundation.
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1192230980_1629146682.85/0/jcl-over-slf4j-1-7-32-sources-jar/org/apache/commons/logging/Log.java
* /opt/cola/permits/1192230980_1629146682.85/0/jcl-over-slf4j-1-7-32-sources-jar/org/apache/commons/logging/LogFactory.java
* /opt/cola/permits/1192230980_1629146682.85/0/jcl-over-slf4j-1-7-32-sources-jar/org/apache/commons/logging/impl/SLF4JLog.java
* /opt/cola/permits/1192230980_1629146682.85/0/jcl-over-slf4j-1-7-32-sources-jar/org/apache/commons/logging/impl/NoOpLog.java
* /opt/cola/permits/1192230980_1629146682.85/0/jcl-over-slf4j-1-7-32-sources-jar/org/apache/commons/logging/impl/SLF4JLocationAwareLog.java
* /opt/cola/permits/1192230980_1629146682.85/0/jcl-over-slf4j-1-7-32-sources-jar/org/apache/commons/logging/impl/SimpleLog.java
* /opt/cola/permits/1192230980_1629146682.85/0/jcl-over-slf4j-1-7-32-sources-jar/org/apache/commons/logging/impl/SLF4JLogFactory.java

No license file was found, but licenses were detected in source scan.

<name>Apache License, Version 2.0</name>

Found in path(s):
* /opt/cola/permits/1192230980_1629146682.85/0/jcl-over-slf4j-1-7-32-sources-jar/META-INF/maven/org.slf4j/jcl-over-slf4j/pom.xml

No license file was found, but licenses were detected in source scan.

2004 The Apache Software Foundation.

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE
Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Found in path(s):
* /opt/cola/permits/1192230980_1629146682.85/0/jcl-over-slf4j-1-7-32-sources-jar/org/apache/commons/logging/package.html

1.208 netty-handler-proxy 4.1.69.Final

1.208.1 Available under license :
No license file was found, but licenses were detected in source scan.

~ Copyright 2014 The Netty Project
~
~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~
~ https://www.apache.org/licenses/LICENSE-2.0
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.

Found in path(s):
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014 The Netty Project
 * 
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * 
 * https://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
1.209 curator-recipes 5.1.0

1.209.1 Available under license:

Curator Recipes
Copyright 2011-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed...
as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.210 jetty-http 9.4.44.v20210927
This program and the accompanying materials are made available under the terms of the Eclipse Public License 2.0 which is available at http://www.eclipse.org/legal/epl-2.0, or the Apache Software License 2.0 which is available at https://www.apache.org/licenses/LICENSE-2.0.

Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

“Contribution” means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and
b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution ‘originates’ from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor’s behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

“Contributor” means any person or entity that distributes the Program.

“Licensed Patents” mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly
perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient’s responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and
b) its license agreement:
   i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
   ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;
   iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and
   iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.
When the Program is made available in source code form:

a) it must be made available under this Agreement; and
b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must:

a) promptly notify the Commercial Contributor in writing of such claim, and
b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN
"AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time.
time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.
You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify, 
defend, and hold each Contributor harmless for any liability 
incurred by, or claims asserted against, such Contributor by reason 
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following 
boilerplate notice, with the fields enclosed by brackets "[]" 
replaced with your own identifying information. (Don't include 
the brackets!) The text should be enclosed in the appropriate 
comment syntax for the file format. We also recommend that a 
file or class name and description of purpose be included on the 
same "printed page" as the copyright notice for easier 
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); 
you may not use this file except in compliance with the License. 
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software 
distributed under the License is distributed on an "AS IS" BASIS, 
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. 
See the License for the specific language governing permissions and 
limitations under the License.

==============================================================
Jetty Web Container
Copyright 1995-2018 Mort Bay Consulting Pty Ltd.
==============================================================

The Jetty Web Container is Copyright Mort Bay Consulting Pty Ltd 
unless otherwise noted.

Jetty is dual licensed under both

* The Apache 2.0 License
  http://www.apache.org/licenses/LICENSE-2.0.html

and

* The Eclipse Public 1.0 License
Jetty may be distributed under either license.

-----

Eclipse

The following artifacts are EPL.
* org.eclipse.jetty.orbit:org.eclipse.jdt.core

The following artifacts are EPL and ASL2.
* org.eclipse.jetty.orbit:javax.security.auth.message

The following artifacts are EPL and CDDL 1.0.
* org.eclipse.jetty.orbit:javax.mail.glassfish

-----

Oracle

The following artifacts are CDDL + GPLv2 with classpath exception.
https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

* javax.servlet:javax.servlet-api
* javax.annotation:javax.annotation-api
* javax.transaction:javax.transaction-api
* javax.websocket:javax.websocket-api

-----

Oracle OpenJDK

If ALPN is used to negotiate HTTP/2 connections, then the following artifacts may be included in the distribution or downloaded when ALPN module is selected.

* java.sun.security.ssl

These artifacts replace/modify OpenJDK classes. The modifications are hosted at github and both modified and original are under GPL v2 with classpath exceptions.
http://openjdk.java.net/legal/gplv2+ce.html

-----

OW2

The following artifacts are licensed by the OW2 Foundation according to the terms of http://asm.ow2.org/license.html
org.ow2.asm:asm-commons
org.ow2.asm:asm

-----
Apache

The following artifacts are ASL2 licensed.

org.apache.taglibs:taglibs-standard-spec
org.apache.taglibs:taglibs-standard-impl

-----
MortBay

The following artifacts are ASL2 licensed. Based on selected classes from following Apache Tomcat jars, all ASL2 licensed.

org.mortbay.jasper:apache-jsp
org.apache.tomcat:tomcat-jasper
org.apache.tomcat:tomcat-juli
org.apache.tomcat:tomcat-jsp-api
org.apache.tomcat:tomcat-el-api
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-api
org.apache.tomcat:tomcat-util-scan
org.apache.tomcat:tomcat-util

org.mortbay.jasper:apache-el
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-el-api

-----
Mortbay

The following artifacts are CDDL + GPLv2 with classpath exception.

https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

org.eclipse.jetty.toolchain:jetty-schemas

-----
Assorted

The UnixCrypt.java code implements the one way cryptography used by
Unix systems for simple password protection. Copyright 1996 Aki Yoshida, modified April 2001 by Iris Van den Broeke, Daniel Deville. Permission to use, copy, modify and distribute UnixCrypt for non-commercial or commercial purposes and without fee is granted provided that the copyright notice appears in all copies.

1.211 netty-codec-stomp 4.1.69.Final

1.211.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 *   https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

No license file was found, but licenses were detected in source scan.
1.212 swagger 1.5.21

1.213 netty-transport-sctp 4.1.69.Final

1.213.1 Available under license:
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2013 The Netty Project
*/
* The Netty Project licenses this file to you under the Apache License, version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at:

* https://www.apache.org/licenses/LICENSE-2.0

* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

*/

Found in path(s):
* /opt/cola/permits/1235364901_1638382290.23//netty-transport-sctp-4-1-69-final-sources-jar/io/netty/channel/sctp/SctpChannelOption.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2012 The Netty Project
* *
* The Netty Project licenses this file to you under the Apache License, version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at:

* https://www.apache.org/licenses/LICENSE-2.0

* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

*/

Found in path(s):
* /opt/cola/permits/1235364901_1638382290.23//netty-transport-sctp-4-1-69-final-sources-jar/io/netty/channel/sctp/oio/package-info.java
* /opt/cola/permits/1235364901_1638382290.23//netty-transport-sctp-4-1-69-final-sources-jar/io/netty/channel/sctp/oio/OioSctpServerChannel.java
* /opt/cola/permits/1235364901_1638382290.23//netty-transport-sctp-4-1-69-final-sources-jar/io/netty/channel/sctp/OioSctpChannel.java
* /opt/cola/permits/1235364901_1638382290.23//netty-transport-sctp-4-1-69-final-sources-jar/io/netty/channel/sctp/nio/package-info.java
* /opt/cola/permits/1235364901_1638382290.23//netty-transport-sctp-4-1-69-final-sources-jar/io/netty/channel/sctp/nio/package-info.java
* /opt/cola/permits/1235364901_1638382290.23//netty-transport-sctp-4-1-69-final-sources-jar/io/netty/channel/sctp/nio/package-info.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2012 The Netty Project
 * 
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * 
 * https://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2011 The Netty Project
 * 
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * 
 * https://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1235364901_1638382290.23/0/netty-transport-sctp-4-1-69-final-sources-jar/io/netty/channel/sctp/SctpChannelConfig.java
No license file was found, but licenses were detected in source scan.

~ Copyright 2012 The Netty Project
~
~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~
~ https://www.apache.org/licenses/LICENSE-2.0
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.

Found in path(s):
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2011 The Netty Project
 * *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * *
 * https://www.apache.org/licenses/LICENSE-2.0
 * *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
  * /opt/cola/permits/1235364901_1638382290.23/0/netty-transport-sctp-4-1-69-final-sources-jar/io/netty/channel/sctp/DefaultSctpServerChannelConfig.java

1.214 spring-tx 5.3.12
1.214.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2018 the original author or authors.
 * 
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 *   https://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/support/ResourceTransactionDefinition.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/jca/cci/connection/ConnectionFactoryUtils.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/jca/support/LocalConnectionFactoryBean.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/jca/cci/CannotCreateRecordException.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/jca/cci/object/MappingCommAreaOperation.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/support/TransactionTemplate.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/jca/cci/core/CciTemplate.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/jca/endpoint/AbstractMessageEndpointFactory.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/jca/cci/core/support/CciDaoSupport.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/jta/WebLogicJtaTransactionManager.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2002-2014 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/jca/context/BootstrapContextAware.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/dao/support/DaoSupport.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/support/DefaultTransactionDefinition.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2002-2015 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
*/
* Copyright 2002-2008 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/support/ResourceHolder.java
No license file was found, but licenses were detected in source scan.

*/
* Copyright 2002-2016 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/dao/support/PersistenceExceptionTranslator.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/jca/cci/core/RecordCreator.java
No license file was found, but licenses were detected in source scan.
/*
 * Copyright 2002-2012 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/jta/ManagedTransactionAdapter.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/dao/TransientDataAccessResourceException.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/jta/JtaAfterCompletionSynchronization.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/jta/UserTransactionAdapter.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/jta/TransactionFactory.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/jca/cci/InvalidResultSetAccessException.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/dao/CannotAcquireLockException.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/interceptor/DelegatingTransactionAttribute.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/interceptor/NoRollbackRuleAttribute.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/jca/cci/connection/NotSupportedRecordFactory.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/IllegalTransactionStateException.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/interceptor/NoRollbackRuleAttribute.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/jca/cci/connection/NotSupportedRecordFactory.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/IllegalTransactionStateException.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/interceptor/NoRollbackRuleAttribute.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/jca/cci/connection/NotSupportedRecordFactory.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/IllegalTransactionStateException.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/interceptor/NoRollbackRuleAttribute.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/jca/cci/connection/NotSupportedRecordFactory.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/IllegalTransactionStateException.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/interceptor/NoRollbackRuleAttribute.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/jca/cci/connection/NotSupportedRecordFactory.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/IllegalTransactionStateException.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/interceptor/NoRollbackRuleAttribute.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/jca/cci/connection/NotSupportedRecordFactory.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/IllegalTransactionStateException.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/interceptor/NoRollbackRuleAttribute.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/jca/cci/connection/NotSupportedRecordFactory.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/IllegalTransactionStateException.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/interceptor/NoRollbackRuleAttribute.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/jca/cci/connection/NotSupportedRecordFactory.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/IllegalTransactionStateException.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/interceptor/NoRollbackRuleAttribute.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/jca/cci/connection/NotSupportedRecordFactory.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/IllegalTransactionStateException.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/interceptor/NoRollbackRuleAttribute.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/jca/cci/connection/NotSupportedRecordFactory.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/IllegalTransactionStateException.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/interceptor/NoRollbackRuleAttribute.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/jca/cci/connection/NotSupportedRecordFactory.java
*/
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2013 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *  
 *  https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/jca/cci/connection/ConnectionSpecConnectionFactoryAdapter.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/interceptor/TransactionAttributeSourceAdvisor.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2017 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *  
 *  https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/jca/cci/connection/ConnectionSpecConnectionFactoryAdapter.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/interceptor/TransactionAttributeSourceAdvisor.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-
jar/org/springframework/jca/support/SimpleBootstrapContext.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-
jar/org/springframework/jca/ccicore/CciOperations.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-
jar/org/springframework/jca/ccicore/connection/DelegatingConnectionFactory.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-
jar/org/springframework/jca/transaction/TransactionSystemException.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-
jar/org/springframework/dao/NonTransientDataAccessException.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-
jar/org/springframework/jca/support/ResourceAdapterFactoryBean.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-
jar/org/springframework/jca/context/BootstrapContextAwareProcessor.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-
jar/org/springframework/delegates/TransactionCallbackWithoutResult.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-
jar/org/springframework/transaction/support/TransactionCallbackWithoutResult.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-
jar/org/springframework/jca/ccicore/ObjectOperations.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-
jar/org/springframework/transaction/support/TransactionCallbackWithoutResult.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-
jar/org/springframework/delegates/CallbackPreferringPlatformTransactionManager.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2002-2020 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
*/
* You may obtain a copy of the License at
  *
  *      https://www.apache.org/licenses/LICENSE-2.0
  *
  * Unless required by applicable law or agreed to in writing, software
  * distributed under the License is distributed on an "AS IS" BASIS,
  * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
  * See the License for the specific language governing permissions and
  * limitations under the License.
  */

Found in path(s):
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/jca/endpoint/GenericMessageEndpointFactory.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/event/TransactionalApplicationListenerSynchronization.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/jca/endpoint/GenericMessageEndpointManager.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/support/TransactionalApplicationListenerAdapter.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/interceptor/MatchAlwaysTransactionAttributeSource.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/config/TxAdviceBeanDefinitionParser.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/interceptor/ProxyTransactionManagementConfiguration.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/interceptor/TransactionAttributeSourcePointcut.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/interceptor/MethodMapTransactionAttributeSource.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/dao/support/DataAccessUtils.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/event/TransactionalEventListenerFactory.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/annotation/EnableTransactionManagement.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/jca/cci/connection/SingleConnectionFactory.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/reactive/TransactionSynchronizationUtils.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2021 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *      https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/support/TransactionSynchronizationUtils.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/event/TransactionalEventListener.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/reactive/TransactionSynchronizationUtils.java
jar/org/springframework/transaction/event/TransactionApplicationListenerMethodAdapter.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/reactive/TransactionContext.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/support/TransactionCallback.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/support/TransactionSynchronizationManager.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/event/TransactionPhase.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/interceptor/DefaultTransactionAttribute.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/reactive/TransactionContextManager.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/annotation/Transactionable.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/interceptor/RuleBasedTransactionAttribute.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/reactive/TransactionCallback.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/reactive/TransactionSynchronizationManager.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/reactive/AbstractReactiveTransactionManager.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/event/TransactionApplicationListener.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2019 the original author or authors.
 * 
 * Licensed under the Apache License, Version 2.0 (the "License);
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 *     https://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/support/AbstractTransactionStatus.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/reactive/GenericReactiveTransaction.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/reactive/TransactionContextHolder.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/support/DefaultTransactionStatus.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/reactive/TransactionalOperatorImpl.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/support/WithoutTransactionOperations.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/reactive/TransactionDefinition.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/reactive/ReactiveResourceSynchronization.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/interceptor/CompositeTransaction/AttributeSource.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/interceptor/TransactionProxyFactoryBean.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/interceptor/TransactionAttributeSource.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/TransactionManager.java
* /opt/cola/permits/1220178478_1635319486.01/0/spring-tx-5-3-12-sources-jar/org/springframework/transaction/support/TransactionOperations.java

Open Source Used In security_automation identity-guest-cs 2176
1.215 jakarta-validation-api 2.0.2

1.215.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Jakarta Bean Validation API
 * 
 * License: Apache License, Version 2.0
 * 
 */

Found in path(s):
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-jar/javax/validation/Validator.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-jar/javax/validation/metadata/CascadableDescriptor.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-jar/javax/validation/ConstraintTarget.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-jar/javax/validation/constraints/ConstraintValidatorContext.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-jar/javax/validation/constraintvalidation/ValidationTarget.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-jar/javax/validation/messages/MessageInterpolator.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-jar/javax/validation/metadata/package-info.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-jar/javax/validation/valueextraction/Unwrapping.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-jar/javax/validation/valueextraction/ValueExtractor.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-jar/javax/validation/constraintvalidation/ValidationTarget.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-jar/javax/validation/metadata/MethodDescriptor.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-jar/javax/validation/metadata/ExecutableDescriptor.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-jar/javax/validation/metadata/ConstraintTarget.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-jar/javax/validation/ConstraintTarget.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-jar/javax/validation/MessageInterpolator.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-jar/javax/validation/ConstraintValidatorContext.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-jar/javax/validation/metadata/ExecutableDescriptor.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-jar/javax/validation/metadata/ExecutableDescriptor.java
*/
No license file was found, but licenses were detected in source scan.

~ Jakarta Bean Validation API
~
~ License: Apache License, Version 2.0
~ See the license.txt file in the root directory or <http://www.apache.org/licenses/LICENSE
<![CDATA[
Comments to: <a href="mailto:bean-validation-dev@eclipse.org">bean-validation-dev@eclipse.org</a>.<br>Copyright &©2019; 2019 Eclipse Foundation.<br>Use is subject to <a href="[ @docRoot]/doc-files/speclicense.html" target="_top">EFSL</a>; this spec is based on material that is licensed under the Apache License, version 2.0.]]>

Found in path(s):
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-jar/META-INF/maven/jakarta.validation/jakarta.validation-api/pom.xml

1.216 spring-aop 5.3.12

1.216.1 Available under license:
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2002-2017 the original author or authors.
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
*      https://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/springframework/aop/support/DelegatePerTargetObjectIntroductionInterceptor.java
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/springframework/aop/interceptor/PerformanceMonitorInterceptor.java
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/springframework/aop/support/DefaultPointcutAdvisor.java
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/springframework/aop/aspectj/AspectInstanceFactory.java
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/springframework/aop/support/AbstractPointcutAdvisor.java
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/springframework/aop/support/DefaultBeanFactoryPointcutAdvisor.java
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/springframework/aop/support/AbstractGenericPointcutAdvisor.java
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/springframework/aop/config/MethodLocatingFactoryBean.java
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/springframework/aop/config/AspectComponentDefinition.java
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/springframework/aop/interceptor/JamonPerformanceMonitorInterceptor.java
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/springframework/aop/framework/autoproxy/InfrastructureAdvisorAutoProxyCreator.java
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/springframework/aop/config/AbstractInterceptorDrivenBeanDefinitionDecorator.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2015 the original author or authors.
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/springframework/aop/aspectj/annotation/PrototypeAspectInstanceFactory.java
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/springframework/aop/aspectj/annotation/AspectJAdvisorFactory.java
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/springframework/aop/ProxyMethodInvocation.java
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/springframework/aop/aspectj/AspectJBeforeAdvice.java
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/springframework/aop/aspectj/SingletonAspectInstanceFactory.java
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/springframework/aop/TargetClassAware.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2016 the original author or authors.
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/springframework/aop/config/AspectJAutoProxyBeanDefinitionParser.java
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/springframework/aop/aspectj/AspectJAfterReturningAdvice.java
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/aopalliance/intercept/MethodInvocation.java
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/aopalliance/intercept/Invocation.java
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/springframework/aop/framework/ProxyFactory.java
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/springframework/aop/framework/autoproxy/TargetSourceCreator.java
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/aopalliance/intercept/ConstructorInterceptor.java
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/springframework/aop/framework/aop/target/ThreadLocalTargetSource.java
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/springframework/aop/support/Pointcuts.java

No license file was found, but licenses were detected in source scan.

*/
* Copyright 2002-2019 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/springframework/aop/support/Pointcuts.java
*/
* Copyright 2002-2012 the original author or authors.
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
* https://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* 
No license file was found, but licenses were detected in source scan.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/springframework/aop/framework/adapter/AdvisorAdapterRegistrationManager.java
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/springframework/aop/framework/AopProxyFactory.java
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/springframework/aop/framework/adapter/AdvisorAdapter.java
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/springframework/aop/target/LazyInitTargetSource.java
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/springframework/aop/target/SimpleBeanTargetSource.java
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/springframework/aop/support/StaticMethodMatcherPointcut.java
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/springframework/aop/aspectj/AspectJAopUtils.java
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/springframework/aop/config/ScopedProxyBeanDefinitionDecorator.java
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/springframework/aop/aspectj/AspectJAopUtils.java
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/springframework/aop/config/ScopedProxyBeanDefinitionDecorator.java
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/springframework/aop/portal/PortalSupportValue.java
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/springframework/aop/annotation/InvokeAdvisorAdviceAdvisorConfigBeanDefinitionParser.java
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/springframework/aop/annotation/InvokeAdvisorAdviceAdvisorConfigBeanDefinitionParser.java
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/springframework/aop/annotation/InvokeAdvisorAdviceAdvisorConfigBeanDefinitionParser.java
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/springframework/aop/annotation/InvokeAdvisorAdviceAdvisorConfigBeanDefinitionParser.java
jar/org/springframework/aop/IntroductionAdvisor.java
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/springframework/aop/support/JdkRegexpMethodPointcut.java
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/springframework/aop/framework/AopProxy.java
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/springframework/aop/support/DynamicMethodMatcherPointcut.java
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/springframework/aop/framework/Refreshable.java
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/springframework/aop/framework/BeanFactoryRefreshableTargetSource.java
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/springframework/aop/framework/PoolConfig.java
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/springframework/aop/framework/framework/adapter/AfterReturningAdviceAdapter.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2002-2021 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*     https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/springframework/aop/aspectj/AspectJAdviceParameterNameDiscoverer.java
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/springframework/aop/aspectj/MethodInvocationProceedingJoinPoint.java
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/springframework/aop/framework/ProxyConfig.java
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/springframework/aop/framework/autoproxy/AbstractAutoProxyCreator.java
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/springframework/aop/support/DefaultIntroductionAdvisor.java
Found in path(s):
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/springframework/aop/framework/DefaultAopProxyFactory.java
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/aopalliance/intercept/Joinpoint.java
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/springframework/aop/framework/AopProxyUtils.java
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/springframework/aop/framework/AbstractAdvisingBeanPostProcessor.java
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/springframework/aop/aspectj/annotation/AbstractAspectJAdvisorFactory.java
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/springframework/aop/framework/InterceptorAndDynamicMethodMatcher.java
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/springframework/aop/target/AbstractPoolingTargetSource.java
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/springframework/aop/aspectj/AspectJExpressionPointcut.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2007 the original author or authors.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
jar/org/springframework/aop/IntroductionInterceptor.java
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-
jar/org/springframework/aop/AfterAdvice.java
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-
jar/org/springframework/aop/SpringProxy.java
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-
jar/org/springframework/aop/framework/AopInfrastructureBean.java
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-
jar/org/springframework/aop/framework/AdvisedSupportListener.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2008 the original author or authors.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * https://www.apache.org/licenses/LICENSE-2.0
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-
jar/org/springframework/aop/ThrowsAdvice.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2006 the original author or authors.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * https://www.apache.org/licenses/LICENSE-2.0
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
Found in path(s):
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/springframework/aop/aspectj/InstantiationModelAwarePointcutAdvisor.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2002-2013 the original author or authors.
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
  *
  *   https://www.apache.org/licenses/LICENSE-2.0
  *
  * Unless required by applicable law or agreed to in writing, software
  * distributed under the License is distributed on an "AS IS" BASIS,
  * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
  * See the License for the specific language governing permissions and
  * limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/springframework/aop/aspectj/AspectJWeaverMessageHandler.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2002-2020 the original author or authors.
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
  *
  *   https://www.apache.org/licenses/LICENSE-2.0
  *
  * Unless required by applicable law or agreed to in writing, software
  * distributed under the License is distributed on an "AS IS" BASIS,
  * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
  * See the License for the specific language governing permissions and
  * limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/springframework/aop/framework/adapter/ThrowsAdviceInterceptor.java
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-jar/org/springframework/aop/aspectj/AspectJAroundAdvice.java
* /opt/cola/permits/1220178487_1635318387.81/0/spring-aop-5-3-12-sources-
/*
* Copyright 2002-2014 the original author or authors.
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
* https://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2002-2018 the original author or authors.
* 
* Licensed under the Apache License, Version 2.0 (the "License");
*/
1.217 audience-annotations 0.5.0

1.217.1 Available under license:

Apache Yetus - Audience Annotations
Copyright 2015-2017 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.
"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."
"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
1.218 scala 2.13.1

1.218.1 Available under license:

Scala includes the JLine library:

Copyright (c) 2002-2006, Marc Prud'hommeaux <mwp1@cornell.edu>
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of JLine nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Scala
Copyright (c) 2002-2019 EPFL
Copyright (c) 2011-2019 Lightbend, Inc.

Scala includes software developed at LAMP/EPFL (https://lamp.epfl.ch/) and Lightbend, Inc. (https://www.lightbend.com/).

Licensed under the Apache License, Version 2.0 (the "License").
Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

This software includes projects with other licenses -- see `doc/LICENSE.md`.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and limitations under the License.
Scala includes the ASM library.

Copyright (c) 2000-2011 INRIA, France Telecom
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of the copyright holders nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

not-a-legal-formal-parameter-tuple.scala:2: error: not a legal formal parameter.
Note: Tuples cannot be directly destructured in method or function parameters.
Either create a single parameter accepting the Tuple2,
or consider a pattern matching anonymous function: `\{( a, b) \Rightarrow ... \}
val x: ((Int, Int) => Int) = (((a, b)) => a)

^ not-a-legal-formal-parameter-tuple.scala:3: error: not a legal formal parameter.
Note: Tuples cannot be directly destructured in method or function parameters.
Either create a single parameter accepting the Tuple2,
or consider a pattern matching anonymous function: `\{( param1, param2) \Rightarrow ... \}
val y: ((Int, Int, Int) => Int) = (((a, !!)) => a)

^ not-a-legal-formal-parameter-tuple.scala:4: error: not a legal formal parameter.
Note: Tuples cannot be directly destructured in method or function parameters.
Either create a single parameter accepting the Tuple3,
or consider a pattern matching anonymous function: `\{( param1, ..., param3) \Rightarrow ... \}
val z: ((Int, Int, Int) => Int) = (((a, NotAPatternVariableName, c)) => a)

^ three errors found
(c) 2012-2014 GitHub

When using the GitHub logos, be sure to follow the GitHub logo guidelines (https://github.com/logos)

Font License: SIL OFL 1.1 (http://scripts.sil.org/OFL)
Applies to all font files

Code License: MIT (http://choosealicense.com/licenses/mit/)
Applies to all other files
Copyright (c) 2006, Ivan Sagalaev
All rights reserved.
Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright
  notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.
* Neither the name of highlight.js nor the names of its contributors
  may be used to endorse or promote products derived from this software
  without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS'' AND ANY
EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE REGENTS AND CONTRIBUTORS BE LIABLE FOR ANY
DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.
Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
Scala is licensed under the [Apache License Version 2.0](https://www.apache.org/licenses/LICENSE-2.0).

## Scala License

Copyright (c) 2002-2019 EPFL

Copyright (c) 2011-2019 Lightbend, Inc.

All rights reserved.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

# Other Licenses

This software includes projects with the following licenses,
which are also included in the `licenses/` directory:

### [Apache License](http://www.apache.org/licenses/LICENSE-2.0.html)
This license is used by the following third-party libraries:

* jansi

### [BSD License](http://www.opensource.org/licenses/bsd-license.php)
This license is used by the following third-party libraries:
* jline
* scalacheck

### [BSD 3-Clause License](http://opensource.org/licenses/BSD-3-Clause)
This license is used by the following third-party libraries:

* asm

### [MIT License](http://www.opensource.org/licenses/MIT)
This license is used by the following third-party libraries:

* jquery

Copyright (c) 2007-2018, Rickard Nilsson

All rights reserved.

Redistribution and use in source and binary forms, with or without modification,
are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice,
  this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice,
  this list of conditions and the following disclaimer in the documentation
  and/or other materials provided with the distribution.
* Neither the name of the EPFL nor the names of its contributors
  may be used to endorse or promote products derived from this software
  without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR
CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR
PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF
LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Scala includes the jQuery library:

Copyright (c) 2010 John Resig
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

(The MIT License)
Copyright (c) 2013 Greg Allen
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.219 antlr 4.5

1.220 netty 4.1.69.Final
1.220.1 Available under license :
Copyright (c) 2003-2008 Yuta Mori All Rights Reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The person or persons who have associated work with this document (the "Dedicator" or "Certifier") hereby either (a) certifies that, to the best of his knowledge, the work of authorship identified is in the public domain of the country from which the work is published, or (b) hereby dedicates whatever copyright the dedicators holds in the work of authorship identified below (the "Work") to the public domain. A certifier, moreover, dedicates any copyright interest he may have in the associated work, and for these purposes, is described as a "dedicator" below.

A certifier has taken reasonable steps to verify the copyright status of this work. Certifier recognizes that his good faith efforts may not shield him from liability if in fact the work certified is not in the public domain.

Dedicator makes this dedication for the benefit of the public at large and to the detriment of the Dedicator's heirs and successors. Dedicator intends this dedication to be an overt act of relinquishment in perpetuity of all present and future rights under copyright law, whether vested or contingent, in the Work. Dedicator understands that such relinquishment of all rights includes the relinquishment of all rights to enforce (by lawsuit or otherwise) those copyrights in the Work.

Dedicator recognizes that, once placed in the public domain, the Work may be freely reproduced, distributed, transmitted, used, modified, built upon, or otherwise exploited by anyone for any purpose, commercial or non-commercial, and in any way, including by methods that have not yet been invented or conceived.

Apache License
Version 2.0, January 2004
https://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.
"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

Apache License
Version 2.0, January 2004
https://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.
You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify, 
defend, and hold each Contributor harmless for any liability 
incurred by, or claims asserted against, such Contributor by reason 
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following 
boilerplate notice, with the fields enclosed by brackets "[]" 
replaced with your own identifying information. (Don't include 
the brackets!) The text should be enclosed in the appropriate 
comment syntax for the file format. We also recommend that a 
file or class name and description of purpose be included on the 
same "printed page" as the copyright notice for easier 
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

   https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software 
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and 
limitations under the License.

The MIT License

Copyright (c) 2009 William Kinney

Permission is hereby granted, free of charge, to any person 
obtaining a copy of this software and associated documentation 
files (the "Software"), to deal in the Software without 
restriction, including without limitation the rights to use, 
copy, modify, merge, publish, distribute, sublicense, and/or sell 
copies of the Software, and to permit persons to whom the 
Software is furnished to do so, subject to the following 
conditions:

The above copyright notice and this permission notice shall be 
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESSION OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Apache License
Version 2.0, January 2004
https://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

Copyright 2009-2010 Ning, Inc.

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

/*
 * Copyright (c) 2004-2006, 2008, 2009, 2011 Apple Inc. All rights reserved.
 *
 * @APPLE_LICENSE_HEADER_START@
 *
 * This file contains Original Code and/or Modifications of Original Code
 * as defined in and that are subject to the Apple Public Source License
 * Version 2.0 (the 'License'). You may not use this file except in
 * compliance with the License. Please obtain a copy of the License at
 * https://www.opensource.apple.com/apsl/ and read it before using this
 * file.
 *
 * The Original Code and all software distributed under the License are
 * distributed on an 'AS IS' basis, WITHOUT WARRANTY OF ANY KIND, EITHER
 * EXPRESS OR IMPLIED, AND APPLE HEREBY DISCLAIMS ALL SUCH WARRANTIES,
 * INCLUDING WITHOUT LIMITATION, ANY WARRANTIES OF MERCHANTABILITY,
 * FITNESS FOR A PARTICULAR PURPOSE, QUIET ENJOYMENT OR NON-INFRINGEMENT.
 * Please see the License for the specific language governing rights and
 * limitations under the License.
 *
 * @APPLE_LICENSE_HEADER_END@
 */

The MIT License (MIT)

Copyright (c) 2014 Cory Benfield

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

Copyright 2011, Google Inc.
All rights reserved.
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright (c) 2010-2011 Matthew J. Francis and Contributors of the jBzip2 Project

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The person or persons who have associated work with this document (the "Dedicator" or "Certifier") hereby either (a) certifies that, to the best of his knowledge, the work of authorship identified is in the public domain of
the country from which the work is published, or (b) hereby dedicates whatever copyright the dedicators holds in the work of authorship identified below (the "Work") to the public domain. A certifier, moreover, dedicates any copyright interest he may have in the associated work, and for these purposes, is described as a "dedicator" below.

A certifier has taken reasonable steps to verify the copyright status of this work. Certifier recognizes that his good faith efforts may not shield him from liability if in fact the work certified is not in the public domain.

Dedicator makes this dedication for the benefit of the public at large and to the detriment of the Dedicator's heirs and successors. Dedicator intends this dedication to be an overt act of relinquishment in perpetuate of all present and future rights under copyright law, whether vested or contingent, in the Work. Dedicator understands that such relinquishment of all rights includes the relinquishment of all rights to enforce (by lawsuit or otherwise) those copyrights in the Work.

Dedicator recognizes that, once placed in the public domain, the Work may be freely reproduced, distributed, transmitted, used, modified, built upon, or otherwise exploited by anyone for any purpose, commercial or non-commercial, and in any way, including by methods that have not yet been invented or conceived.

Apache License
Version 2.0, January 2004
https://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.
"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
   Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
   stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
   that You distribute, all copyright, patent, trademark, and
   attribution notices from the Source form of the Work,
   excluding those notices that do not pertain to any part of
   the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.
You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
The MIT License

Copyright (c) 2012, 2014, 2015, 2016 Tatsuhiro Tsujikawa
Copyright (c) 2012, 2014, 2015, 2016 nghttp2 contributors

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License. You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

(BSD License: https://www.opensource.org/licenses/bsd-license)

Copyright (c) 2011, Joe Walnes, Aslak Hellesy and contributors
All rights reserved.

Redistribution and use in source and binary forms, with or
without modification, are permitted provided that the
following conditions are met:

* Redistributions of source code must retain the above
copyright notice, this list of conditions and the
following disclaimer.

* Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the
following disclaimer in the documentation and/or other
materials provided with the distribution.

* Neither the name of the Webbit nor the names of
its contributors may be used to endorse or promote products
derived from this software without specific prior written
permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND
CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES,
INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR
CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE
GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR
BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The MIT License (MIT)

Copyright (c) 2000 - 2013 The Legion of the Bouncy Castle Inc.
(https://www.bouncycastle.org)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2000,2001,2002,2003,2004 ymnk, JCraft,Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. The names of the authors may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL JCRAFT, INC. OR ANY CONTRIBUTORS TO THIS SOFTWARE BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA,
The Netty Project

The Netty Project licenses this file to you under the Apache License, version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at:

https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
Also, please refer to each LICENSE.<component>.txt file, which is located in the 'license' directory of the distribution file, for the license terms of the components that this product depends on.

This product contains the extensions to Java Collections Framework which has been derived from the works by JSR-166 EG, Doug Lea, and Jason T. Greene:

* LICENSE:
  * license/LICENSE.jsr166y.txt (Public Domain)
* HOMEPAGE:
  * http://gee.cs.oswego.edu/cgi-bin/viewcvs.cgi/jsr166/
  * http://viewvc.jboss.org/cgi-bin/viewvc.cgi/jbosscache/experimental/jsr166/

This product contains a modified version of Robert Harder's Public Domain Base64 Encoder and Decoder, which can be obtained at:

* LICENSE:
  * license/LICENSE.base64.txt (Public Domain)
* HOMEPAGE:
  * http://iharder.sourceforge.net/current/java/base64/

This product contains a modified portion of 'Webbit', an event based WebSocket and HTTP server, which can be obtained at:

* LICENSE:
  * license/LICENSE.webbit.txt (BSD License)
* HOMEPAGE:
  * https://github.com/joewalnes/webbit

This product contains a modified portion of 'SLF4J', a simple logging facade for Java, which can be obtained at:

* LICENSE:
  * license/LICENSE.slf4j.txt (MIT License)
* HOMEPAGE:
  * https://www.slf4j.org/

This product contains a modified portion of 'Apache Harmony', an open source Java SE, which can be obtained at:

* NOTICE:
  * license/NOTICE.harmony.txt
* LICENSE:
  * license/LICENSE.harmony.txt (Apache License 2.0)
* HOMEPAGE:
  * https://archive.apache.org/dist/harmony/
This product contains a modified portion of 'jbzip2', a Java bzip2 compression and decompression library written by Matthew J. Francis. It can be obtained at:

* LICENSE:  
  * license/LICENSE.jbzip2.txt (MIT License)  
  * HOMEPAGE:  
    * https://code.google.com/p/jbzip2/

This product contains a modified portion of 'libdivsufsort', a C API library to construct the suffix array and the Burrows-Wheeler transformed string for any input string of a constant-size alphabet written by Yuta Mori. It can be obtained at:

* LICENSE:  
  * license/LICENSE.libdivsufsort.txt (MIT License)  
  * HOMEPAGE:  
    * https://github.com/y-256/libdivsufsort

This product contains a modified portion of Nitsan Wakart's 'JCTools', Java Concurrency Tools for the JVM, which can be obtained at:

* LICENSE:  
  * license/LICENSE.jctools.txt (ASL2 License)  
  * HOMEPAGE:  
    * https://github.com/JCTools/JCTools

This product optionally depends on 'JZlib', a re-implementation of zlib in pure Java, which can be obtained at:

* LICENSE:  
  * license/LICENSE.jzlib.txt (BSD style License)  
  * HOMEPAGE:  
    * http://www.jcraft.com/jzlib/

This product optionally depends on 'Compress-LZF', a Java library for encoding and decoding data in LZF format, written by Tatu Saloranta. It can be obtained at:

* LICENSE:  
  * license/LICENSE.compress-lzf.txt (Apache License 2.0)  
  * HOMEPAGE:  
    * https://github.com/ning/compress

This product optionally depends on 'lz4', a LZ4 Java compression and decompression library written by Adrien Grand. It can be obtained at:

* LICENSE:  
  * license/LICENSE.lz4.txt (Apache License 2.0)  
  * HOMEPAGE:
This product optionally depends on 'lz4-java', a LZMA Java compression and decompression library, which can be obtained at:

* LICENSE:
  * license/LICENSE.lz4-java.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/jpountz/lz4-java

This product optionally depends on 'lzma-java', a LZMA Java compression and decompression library, which can be obtained at:

* LICENSE:
  * license/LICENSE.lzma-java.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/lponge/lzma-java

This product optionally depends on 'zstd-jni', a zstd-jni Java compression and decompression library, which can be obtained at:

* LICENSE:
  * license/LICENSE.zstd-jni.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/luben/zstd-jni

This product contains a modified portion of 'jfastlz', a Java port of FastLZ compression and decompression library written by William Kinney. It can be obtained at:

* LICENSE:
  * license/LICENSE.jfastlz.txt (MIT License)
* HOMEPAGE:
  * https://code.google.com/p/jfastlz/

This product contains a modified portion of and optionally depends on 'Protocol Buffers', Google's data interchange format, which can be obtained at:

* LICENSE:
  * license/LICENSE.protobuf.txt (New BSD License)
* HOMEPAGE:
  * https://github.com/google/protobuf

This product optionally depends on 'Bouncy Castle Crypto APIs' to generate a temporary self-signed X.509 certificate when the JVM does not provide the equivalent functionality. It can be obtained at:

* LICENSE:
  * license/LICENSE.bouncycastle.txt (MIT License)
* HOMEPAGE:
  * https://www.bouncycastle.org/

This product optionally depends on 'Snappy', a compression library produced by Google Inc, which can be obtained at:

* LICENSE:
  * license/LICENSE.snappy.txt (New BSD License)
This product optionally depends on 'JBoss Marshalling', an alternative Java serialization API, which can be obtained at:

* LICENSE:
  * license/LICENSE.jboss-marshalling.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/jboss-remoting/jboss-marshalling

This product optionally depends on 'Caliper', Google's micro-benchmarking framework, which can be obtained at:

* LICENSE:
  * license/LICENSE.caliper.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/google/caliper

This product optionally depends on 'Apache Commons Logging', a logging framework, which can be obtained at:

* LICENSE:
  * license/LICENSE.commons-logging.txt (Apache License 2.0)
* HOMEPAGE:
  * https://commons.apache.org/logging/

This product optionally depends on 'Apache Log4J', a logging framework, which can be obtained at:

* LICENSE:
  * license/LICENSE.log4j.txt (Apache License 2.0)
* HOMEPAGE:
  * https://logging.apache.org/log4j/

This product optionally depends on 'Aalto XML', an ultra-high performance non-blocking XML processor, which can be obtained at:

* LICENSE:
  * license/LICENSE.aalto-xml.txt (Apache License 2.0)
* HOMEPAGE:
  * https://wiki.fasterxml.com/AaltoHome

This product contains a modified version of 'HPACK', a Java implementation of the HTTP/2 HPACK algorithm written by Twitter. It can be obtained at:

* LICENSE:
  * license/LICENSE.hpack.txt (Apache License 2.0)
This product contains a modified version of 'HPACK', a Java implementation of the HTTP/2 HPACK algorithm written by Cory Benfield. It can be obtained at:

* LICENSE:
  * license/LICENSE.hyper-hpack.txt (MIT License)
* HOMEPAGE:
  * https://github.com/twitter/hpack

This product contains a modified version of 'HPACK', a Java implementation of the HTTP/2 HPACK algorithm written by Tatsuhiro Tsujikawa. It can be obtained at:

* LICENSE:
  * license/LICENSE.nghttp2-hpack.txt (MIT License)
* HOMEPAGE:
  * https://github.com/nghttp2/nghttp2/

This product contains a modified portion of 'Apache Commons Lang', a Java library provides utilities for the java.lang API, which can be obtained at:

* LICENSE:
  * license/LICENSE.commons-lang.txt (Apache License 2.0)
* HOMEPAGE:
  * https://commons.apache.org/proper/commons-lang/

This product contains the Maven wrapper scripts from 'Maven Wrapper', that provides an easy way to ensure a user has everything necessary to run the Maven build.

* LICENSE:
  * license/LICENSE.mvn-wrapper.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/takari/maven-wrapper

This product contains the dnsinfo.h header file, that provides a way to retrieve the system DNS configuration on MacOS. This private header is also used by Apple's open source mDNSResponder (https://opensource.apple.com/tarballs/mDNSResponder/).

* LICENSE:
  * license/LICENSE.dnsinfo.txt (Apple Public Source License 2.0)
* HOMEPAGE:

This product optionally depends on 'Brotli4j', Brotli compression and decompression for Java., which can be obtained at:
A copy is also included with both the the downloadable source code package and jar that contains class bytecodes, as file "ASL 2.0". In both cases, that file should be located next to this file: in source distribution the location should be "release-notes/asl"; and in jar "META-INF/"
Apache Harmony

Copyright 2006, 2010 The Apache Software Foundation.

This product includes software developed at
The Apache Software Foundation (https://www.apache.org/).

1.221 okhttp 3.14.9
1.221.1 Available under license :
Note that publicsuffixes.gz is compiled from The Public Suffix List:
https://publicsuffix.org/list/public_suffix_list.dat

It is subject to the terms of the Mozilla Public License, v. 2.0:
https://mozilla.org/MPL/2.0/

1.222 txw2 2.3.5
1.222.1 Available under license :
Copyright (c) 2018 Oracle and/or its affiliates. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

- Neither the name of the Eclipse Foundation, Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
This content is produced and maintained by the Eclipse Implementation of JAXB project.

* Project home: https://projects.eclipse.org/projects/ee4j.jaxb-impl

## Trademarks

Eclipse Implementation of JAXB is a trademark of the Eclipse Foundation.

## Copyright

All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed source code repository logs.

## Declared Project Licenses

This program and the accompanying materials are made available under the terms of the Eclipse Distribution License v. 1.0 which is available at http://www.eclipse.org/org/documents/edl-v10.php.

SPDX-License-Identifier: BSD-3-Clause

## Source Code

The project maintains the following source code repositories:

* https://github.com/eclipse-ee4j/jaxb-ri
* https://github.com/eclipse-ee4j/jaxb-istack-commons
* https://github.com/eclipse-ee4j/jaxb-dtd-parser
* https://github.com/eclipse-ee4j/jaxb-fi
* https://github.com/eclipse-ee4j/jaxb-stax-ex
* https://github.com/eclipse-ee4j/jax-rpc-ri

## Third-party Content

This project leverages the following third party content.

Apache Ant (1.10.2)

* License: Apache-2.0 AND W3C AND LicenseRef-Public-Domain
Apache Ant (1.10.2)

* License: Apache-2.0 AND W3C AND LicenseRef-Public-Domain

Apache Felix (1.2.0)

* License: Apache License, 2.0

args4j (2.33)

* License: MIT License

dom4j (1.6.1)

* License: Custom license based on Apache 1.1

file-management (3.0.0)

* License: Apache-2.0
  * Project: https://maven.apache.org/shared/file-management/
  * Source:
    https://svn.apache.org/viewvc/maven/shared/tags/file-management-3.0.0/

JUnit (4.12)

* License: Eclipse Public License

JUnit (4.12)

* License: Eclipse Public License

maven-compat (3.5.2)

* License: Apache-2.0
  * Project: https://maven.apache.org/ref/3.5.2/maven-compat/
  * Source:
    https://mvnrepository.com/artifact/org.apache.maven/maven-compat/3.5.2

maven-core (3.5.2)

* License: Apache-2.0
  * Project: https://maven.apache.org/ref/3.5.2/maven-core/index.html
  * Source: https://mvnrepository.com/artifact/org.apache.maven/maven-core/3.5.2

maven-plugin-annotations (3.5)

* License: Apache-2.0
* Project: https://maven.apache.org/plugin-tools/maven-plugin-annotations/
* Source:
  https://github.com/apache/maven-plugin-tools/tree/master/maven-plugin-annotations

maven-plugin-api (3.5.2)

* License: Apache-2.0

maven-resolver-api (1.1.1)

* License: Apache-2.0

maven-resolver-api (1.1.1)

* License: Apache-2.0

maven-resolver-connector-basic (1.1.1)

* License: Apache-2.0

maven-resolver-impl (1.1.1)

* License: Apache-2.0

maven-resolver-spi (1.1.1)

* License: Apache-2.0

maven-resolver-transport-file (1.1.1)

* License: Apache-2.0
* Project: https://maven.apache.org/resolver/maven-resolver-transport-file/
* Source:
  https://github.com/apache/maven-resolver/tree/master/maven-resolver-transport-file

maven-resolver-util (1.1.1)

* License: Apache-2.0

maven-settings (3.5.2)

* License: Apache-2.0
* Source:
  https://mvnrepository.com/artifact/org.apache.maven/maven-settings/3.5.2

OSGi Service Platform Core Companion Code (6.0)

* License: Apache License, 2.0
plexus-archiver (3.5)

- License: Apache-2.0
- Project: https://codehaus-plexus.github.io/plexus-archiver/
- Source: https://github.com/codehaus-plexus/plexus-archiver

plexus-io (3.0.0)

- License: Apache-2.0

plexus-utils (3.1.0)

- License: Apache-2.0 or Apache- 1.1 or BSD or Public Domain or Indiana University Extreme! Lab Software License V1.1.1 (Apache 1.1 style)

relaxng-datatype (1.0)

- License: New BSD license

Sax (0.2)

- License: SAX-PD
- Source: http://sourceforge.net/project/showfiles.php?group_id=29449

testng (6.14.2)

- License: Apache-2.0 AND (MIT OR GPL-1.0+)
- Project: https://testng.org/doc/index.html
- Source: https://github.com/cbeust/testng

wagon-http-lightweight (3.0.0)

- License: Pending
- Project: https://maven.apache.org/wagon/
- Source:

xz for java (1.8)

- License: LicenseRef-Public-Domain

## Cryptography

Content may contain encryption software. The country in which you are currently
may have restrictions on the import, possession, and use, and/or re-export to
another country, of encryption software. BEFORE using any encryption software,
please check the country's laws, regulations and policies concerning the import, possession, or use, and re-export of encryption software, to see if this is permitted.

1.223 rtx 2.1.7

1.223.1 Available under license:

/*--------------------------------------*/
| RXTX License v 2.1 - LGPL v 2.1 + Linking Over Controlled Interface. |
| RXTX is a native interface to serial ports in java.                  |
| Copyright 1997-2007 by Trent Jarvi tjarvi@qbang.org and others who  |
| actually wrote it. See individual source files for more information. |
|
| A copy of the LGPL v 2.1 may be found at                             |
| http://www.gnu.org/licenses/lgpl.txt on March 4th 2007. A copy is   |
| here for your convenience.                                          |
|
| This library is free software; you can redistribute it and/or       |
| modify it under the terms of the GNU Lesser General Public License |
| as published by the Free Software Foundation; either                |
| version 2.1 of the License, or (at your option) any later version.  |
|
| This library is distributed in the hope that it will be useful,     |
| but WITHOUT ANY WARRANTY; without even the implied warranty of      |
| MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU     |
| Lesser General Public License for more details.                     |
|
| An executable that contains no derivative of any portion of RXTX,   |
| but is designed to work with RXTX by being dynamically linked with  |
| it, is considered a "work that uses the Library" subject to the terms and conditions of the GNU Lesser General Public License. |
|
| The following has been added to the RXTX License to remove          |
| any confusion about linking to RXTX. We want to allow in part what  |
| section 5, paragraph 2 of the LGPL does not permit in the special    |
| case of linking over a controlled interface. The intent is to add a  |
| Java Specification Request or standards body defined interface in the |
| future as another exception but one is not currently available.      |
|
| http://www.fsf.org/licenses/gpl-faq.html#LinkingOverControlledInterface |
|
| As a special exception, the copyright holders of RXTX give you      |
| permission to link RXTX with independent modules that communicate with RXTX solely through the Sun Microsystems CommAPI interface version 2, regardless of the license terms of these independent modules, and to copy and distribute the resulting combined work under terms of your choice, provided that every copy of the combined work is accompanied by a complete
copy of the source code of RXTX (the version of RXTX used to produce the
combined work), being distributed under the terms of the GNU Lesser General
Public License plus this exception. An independent module is a
module which is not derived from or based on RXTX.

Note that people who make modified versions of RXTX are not obligated
to grant this special exception for their modified versions; it is
their choice whether to do so. The GNU Lesser General Public License
gives permission to release a modified version without this exception; this
exception also makes it possible to release a modified version which
carries forward this exception.

You should have received a copy of the GNU Lesser General Public
License along with this library; if not, write to the Free
Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA
All trademarks belong to their respective owners.

The original GNU Lesser General Public License follows.

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts
as the successor of the GNU Library Public License, version 2, hence
the version number 2.1.]

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
Licenses are intended to guarantee your freedom to share and change
free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some
specially designated software packages--typically libraries--of the
Free Software Foundation and other authors who decide to use it. You
can use it too, but we suggest you first think carefully about whether
this license or the ordinary General Public License is the better
strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use,
not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary
General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".
A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.
c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.
Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6.
Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these
materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying
the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.
13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS
How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library "Frob" (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!

1.224 antlr 3.5.2
1.224.1 Available under license:
BSD licence
http://antlr.org/license.html

1.225 spring-data-commons 2.4.14
1.225.1 Available under license:
Spring Data Commons 2.4.14 (2020.0.14)
Copyright (c) [2010-2021] Pivotal Software, Inc.

This product is licensed to you under the Apache License, Version 2.0 (the "License").
You may not use this product except in compliance with the License.

This product may include a number of subcomponents with
separate copyright notices and license terms. Your use of the source
code for the these subcomponents is subject to the terms and
conditions of the subcomponent's license, as noted in the LICENSE file.

Apache License
Version 2.0, January 2004
https://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of Your accepting any such warranty or additional liability.
APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

=======================================================================

To the extent any open source subcomponents are licensed under the EPL and/or other
similar licenses that require the source code and/or modifications to
source code to be made available (as would be noted above), you may obtain a
copy of the source code corresponding to the binaries for such open source
components and modifications thereto, if any, (the "Source Files"), by
downloading the Source Files from https://www.springsource.org/download,
or by sending a request, with your name and address to: VMware, Inc., 3401 Hillview
Avenue, Palo Alto, CA 94304, United States of America or email info@vmware.com. All
such requests should clearly specify: OPEN SOURCE FILES REQUEST, Attention General
Counsel. VMware shall mail a copy of the Source Files to you on a CD or equivalent
physical medium. This offer to obtain a copy of the Source Files is valid for three
years from the date you acquired this Software product.

1.226 jackson-xc 2.13.0
1.226.1 Available under license:

# Jackson JSON processor

Jackson is a high-performance, Free/Open Source JSON processing library. It was originally written by Tatu Saloranta (tatu.saloranta@iki.fi), and has been in development since 2007. It is currently developed by a community of developers, as well as supported commercially by FasterXML.com.

## Licensing

Jackson core and extension components may licensed under different licenses. To find the details that apply to this artifact see the accompanying LICENSE file. For more information, including possible other licensing options, contact FasterXML.com (http://fasterxml.com).

## Credits

A list of contributors may be found from CREDITS file, which is included in some artifacts (usually source distributions); but is always available from the source code management (SCM) system project uses. This copy of Jackson JSON processor `jackson-module-jaxb-annotations` module is licensed under the Apache (Software) License, version 2.0 ("the License"). See the License for details about distribution rights, and the specific rights regarding derivate works.

You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

1.227 spring-context 5.3.12

1.227.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2020 the original author or authors.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * https://www.apache.org/licenses/LICENSE-2.0
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
/**
* Abstract the invocation of a cache operation.
*<p>Does not provide a way to transmit checked exceptions but
* provide a special exception that should be used to wrap any
* exception that was thrown by the underlying invocation.
* Callers are expected to handle this issue type specifically.
*
* @author Stephane Nicoll
* @since 4.1
*/

Found in path(s):
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-jar/org/springframework/cache/interceptor/CacheOperationInvoker.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2002-2012 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-jar/org/springframework/remoting/RemoteConnectFailureException.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-jar/org/springframework/scheduling/SchedulingException.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-jar/org/springframework/ejb/config/JeeNamespaceHandler.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-jar/org/springframework/scheduling/Trigger.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-jar/org/springframework/ui/context/ThemeSource.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2013 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-
  jar/org/springframework/context/i18n/TimeZoneAwareLocaleContext.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-
  jar/org/springframework/context/annotation/FilterType.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-
  jar/org/springframework/format/datetime/joda/LocalTimeParser.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-
  jar/org/springframework/context/i18n/LocaleContext.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-
  jar/org/springframework/context/annotation/Lazy.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-
  jar/org/springframework/context/support/MessageSourceResourceBundle.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-
  jar/org/springframework/format/datetime/joda/ReadablePartialPrinter.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-
  jar/org/springframework/context/annotation/Description.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-
  jar/org/springframework/format/datetime/joda/ReadableInstantPrinter.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-
  jar/org/springframework/scheduling/concurrent/CustomizableThreadFactory.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-
  jar/org/springframework/format/datetime/joda/LocalDateTimeParser.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-
  jar/org/springframework/validation/beanvalidation/OptionalValidatorFactoryBean.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-
  jar/org/springframework/format/datetime/joda/MillisecondInstantPrinter.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-
  jar/org/springframework/format/datetime/joda/LocalDateParser.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2020 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *      https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* Indicates that a component is only eligible for registration when all
  * [@linkplain #value specified conditions] match.
*<p>A <em>condition</em> is any state that can be determined programmatically
  * before the bean definition is due to be registered (see [@link Condition] for details).
*<p>The [@code @Conditional] annotation may be used in any of the following ways:
  * <ul>
  * <li>as a type-level annotation on any class directly or indirectly annotated with
  * [@code @Component], including [@link Configuration @Configuration] classes</li>
  * <li>as a meta-annotation, for the purpose of composing custom stereotype</li>
  * annotations</li>
  * <li>as a method-level annotation on any [@link Bean @Bean] method</li>
  * </ul>
*<p>If a [@code @Configuration] class is marked with [@code @Conditional],
  * all of the [@code @Bean] methods, [@link Import @Import] annotations, and
  * [@link ComponentScan @ComponentScan] annotations associated with that
  * class will be subject to the conditions.
*<p><strong>NOTE</strong>: Inheritance of [@code @Conditional] annotations
  * is not supported; any conditions from superclasses or from overridden
  * methods will not be considered. In order to enforce these semantics,
  * [@code @Conditional] itself is not declared as
  * [@link java.lang.annotation.Inherited @Inherited]; furthermore, any
  * custom <em>composed annotation</em> that is meta-annotated with
  * [@code @Conditional] must not be declared as [@code @Inherited].
*<p>@author Phillip Webb
* @author Sam Brannen
* @since 4.0
* @see Condition
*/

Found in path(s):
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-
  jar/org/springframework/context/annotation/Conditional.java
No license file was found, but licenses were detected in source scan.

*/
* Copyright 2002-2017 the original author or authors
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* https://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-jar/org/springframework/ui/ModelMapExtensions.kt
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-jar/org/springframework/ui/ModelExtensions.kt
No license file was found, but licenses were detected in source scan.

/ *
* Copyright 2002-2020 the original author or authors.
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* https://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-jar/org/springframework/scheduling/concurrent/ReschedulingRunnable.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-jar/org/springframework/context/ApplicationStartupAware.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-jar/org/springframework/validation/BindException.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-jar/org/springframework/context/annotation/ComponentScan.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-jar/org/springframework/context/event/EventPublicationInterceptor.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-jar/org/springframework/context/support/LiveBeansViewMBean.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-
/*
 * Copyright 2002-2018 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *      https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

No license file was found, but licenses were detected in source scan.

* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources.jar/org/springframework/context/annotation/DeferredImportSelector.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources.jar/org/springframework/scheduling/TaskScheduler.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources.jar/org/springframework/context/annotation/AutoProxyRegistrar.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources.jar/org/springframework/scheduling/annotation/SchedulingConfiguration.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources.jar/org/springframework/ejb/access/AbstractRemoteSlsbInvokerInterceptor.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources.jar/org/springframework/context/annotation/ContextAnnotationAutowireCandidateResolver.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources.jar/org/springframework/format/support/DefaultFormattingConversionService.java

Found in path(s):
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources.jar/org/springframework/scheduling/support/ScheduledMethodRunnable.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources.jar/org/springframework/validation/SmartValidator.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources.jar/org/springframework/context/expression/BeanFactoryResolver.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources.jar/org/springframework/scheduling/annotation/EnableAsync.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources.jar/org/springframework/remoting/support/RemoteInvocationResult.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources.jar/org/springframework/ejb/access/AbstractSlsbInvokerInterceptor.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources.jar/org/springframework/validation/validation/BindingResult.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2007 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *   https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-jar/org/springframework/remoting/support/RemoteInvocationFactory.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-jar/org/springframework/jmx/export/metadata/ManagedOperationParameter.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-jar/org/springframework/remoting/support/RemoteInvocationExecutor.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-jar/org/springframework/jmx/export/SpringModelMBean.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2002-2015 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-jar/org/springframework/jmx/export/annotation/ManagedOperation.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-jar/org/springframework/jmx/export/annotation/ManagedResource.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-jar/org/springframework/format/datetime/joda/PeriodFormatter.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-jar/org/springframework/context/event/EventListenerFactory.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-jar/org/springframework/validation/beanvalidation/SpringConstraintValidatorFactory.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2017 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *      https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2002-2016 the original author or authors.
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* https://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-jar/org/springframework/format/datetime/standard/YearMonthFormatter.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-jar/org/springframework/context/support/GenericXmlApplicationContext.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-jar/org/springframework/format/Printer.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-jar/org/springframework/format/datetime/standard/PeriodFormatter.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-jar/org/springframework/format/datetime/standard/TemporalAccessorPrinter.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-jar/org/springframework/context/annotation/Primary.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-jar/org/springframework/context/annotation/ScopeMetadataResolver.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-jar/org/springframework/cache/interceptor/SimpleKeyGenerator.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-jar/org/springframework/scheduling/config/ExecutorBeanDefinitionParser.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-jar/org/springframework/scheduling/config/ExecutorBeanDefinitionParser.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2009 the original author or authors.
 * 
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 * https://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5.3.12-sources-jar/org/springframework/context/LifecycleProcessor.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2005 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *      https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5.3.12-sources-jar/org/springframework/jmx/export/metadata/ManagedOperation.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2021 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *      https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5.3.12-sources-jar/org/springframework/scheduling/support/CronExpression.java
Open Source Used In security_automation identity-guest-cs 2296

* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources.jar/org/springframework/context/PayloadApplicationEvent.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources.jar/org/springframework/context/support/ReloadableResourceBundleMessageSource.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources.jar/org/springframework/context/support/PostProcessorRegistrationDelegate.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources.jar/org/springframework/context/support/PropertySourcesPlaceholderConfigurer.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources.jar/org/springframework/context/ApplicationEvent.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources.jar/org/springframework/context/event/GenericApplicationListenerAdapter.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources.jar/org/springframework/context/event/ApplicationEventMulticaster.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources.jar/org/springframework/validation/support/BindingAwareModelMap.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources.jar/org/springframework/cache/interceptor/CacheOperation.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources.jar/org/springframework/scheduling/annotation/ScheduledAnnotationBeanPostProcessor.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources.jar/org/springframework/context/event/ApplicationListenerMethodAdapter.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources.jar/org/springframework/context/event/SmartApplicationListener.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources.jar/org/springframework/scheduling/support/BitsCronField.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources.jar/org/springframework/format/datetime/standard/DateTimeFormatterFactory.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources.jar/org/springframework/scheduling/annotation/SchedulingConfigurer.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources.jar/org/springframework/format/FormatterRegistrar.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources.jar/org/springframework/context/annotation/ClassPathScanningCandidateComponentProvider.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources.jar/org/springframework/validation/validation/LocalValidatorFactoryBean.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources.jar/org/springframework/format/datetime/standard/DateTimeContextHolder.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources.jar/org/springframework/cache/annotation/Cacheable.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources.jar/org/springframework/context/annotation/ConfigurationClassUtils.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources.jar/org/springframework/scheduling/support/CronTrigger.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources.jar/org/springframework/cache/interceptor/CacheableOperation.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources.jar/org/springframework/context/annotation/EnableAspectJAutoProxy.java
No license file was found, but licenses were detected in source scan.
/*
 * Copyright 2002-2011 the original author or authors.
 * 
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 *   https://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-jar/org/springframework/jmx/export/notification/NotificationPublisherAware.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-jar/org/springframework/context/ApplicationEventPublisherAware.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-jar/org/springframework/context/annotation/ConflictingBeanDefinitionException.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-jar/org/springframework/context/annotation/ImportAware.java

No license file was found, but licenses were detected in source scan.

;/*
 * Copyright 2002-2019 the original author or authors.
 * 
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 *   https://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-jar/org/springframework/ui/Model.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-jar/org/springframework/scheduling/support/TaskUtils.java

---

Open Source Used In security_automation identity-guest-cs 2298
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2014 the original author or authors.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 * Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-
  jar/org/springframework/validation/beanvalidation/MessageSourceResourceBundleLocator.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-
  jar/org/springframework/context/ApplicationContext.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-
  jar/org/springframework/scheduling/annotation/AsyncConfigurerSupport.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-
  jar/org/springframework/cache/interceptor/BasicOperation.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-
  jar/org/springframework/scheduling/config/ContextLifecycleScheduledTaskRegistrar.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-
  jar/org/springframework/validation/annotation/Validated.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-
  jar/org/springframework/format/AnnotationFormatterFactory.java
* /opt/cola/permits/1220178469_1635319037.6/0/spring-context-5-3-12-sources-
  jar/org/springframework/context/annotation/PropertySources.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2002-2019 the original author or authors.
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
*     https://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
/**
* {@link FactoryBean} that obtains a WebSphere {@link javax.management.MBeanServer}
* reference through WebSphere's proprietary {@code AdminServiceFactory} API,
* available on WebSphere 5.1 and higher.
* 
* <p>Exposes the {@code MBeanServer} for bean references.
* 
* <p>This {@code FactoryBean} is a direct alternative to {@link MBeanServerFactoryBean},

---

Open Source Used In security_automation identity-guest-cs 2301
which uses standard JMX 1.2 API to access the platform's \{@link MBeanServer\}.

See the javadocs for WebSphere's

and

@code AdminServiceFactory
@code MBeanFactory

@author Juergen Hoeller
@author Rob Harrop
@since 2.0.3
@see javax.management.MBeanServer
@see MBeanServerFactoryBean

1.228 jacoco-agent 0.8.5
1.228.1 Available under license:

   Found license 'Eclipse Public License 1.0' in 'This program and the accompanying materials are made available under the terms of the Eclipse Public License 2.0 which is available at'

   Found license 'Eclipse Public License 1.0' in 'under the terms and conditions of the Eclipse Public License Version 2.0'

1.229 jaxb-runtime 2.3.0
1.229.1 Available under license:

   COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.1

1. Definitions.

1.1. "Contributor" means each individual or entity that creates or contributes to the creation of Modifications.

1.2. "Contributor Version" means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. "Covered Software" means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including
portions thereof.

1.4. "Executable" means the Covered Software in any form other than Source Code.

1.5. "Initial Developer" means the individual or entity that first makes Original Software available under this License.

1.6. "Larger Work" means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. "License" means this document.

1.8. "Licensable" means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. "Modifications" means the Source Code and Executable form of any of the following:

A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

B. Any new file that contains any part of the Original Software or previous Modification; or

C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. "Original Software" means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. "Patent Claims" means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. "Source Code" means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. "You" (or "Your") means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, "You" includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity,
whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling
of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any
3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients' rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.


You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.

Oracle is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be
given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABILITY, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the
termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as "Participant") alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. If You assert a patent infringement claim against Participant alleging that the Participant Software directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

6.4. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY’S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR
8. U.S. GOVERNMENT END USERS.

The Covered Software is a "commercial item," as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of "commercial computer software" (as that term is defined at 48 C.F.R. § 252.227-7014(a)(1)) and "commercial computer software documentation" as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.
/*
 * DO NOT ALTER OR REMOVE COPYRIGHT NOTICES OR THIS HEADER.
 *
 * Copyright (c) 1997-2017 Oracle and/or its affiliates. All rights reserved.
 *
 * The contents of this file are subject to the terms of either the GNU
 * General Public License Version 2 only ("GPL") or the Common Development
 * and Distribution License("CDDL") (collectively, the "License"). You
 * may not use this file except in compliance with the License. You can
 * obtain a copy of the License at
 * https://oss.oracle.com/licenses/CDDL+GPL-1.1
 * or LICENSE.txt. See the License for the specific
 * language governing permissions and limitations under the License.
 *
 * When distributing the software, include this License Header Notice in each
 * file and include the License file at LICENSE.txt.
 *
 * GPL Classpath Exception:
 * Oracle designates this particular file as subject to the "Classpath"
 * exception as provided by Oracle in the GPL Version 2 section of the License
 * file that accompanied this code.
 *
 * Modifications:
 * If applicable, add the following below the License Header, with the fields
 * enclosed by brackets [] replaced by your own identifying information:
 * "Portions Copyright [year] [name of copyright owner]"
 *
 * Contributor(s):
 * If you wish your version of this file to be governed by only the CDDL or
 * only the GPL Version 2, indicate your decision by adding "[Contributor]
 * elects to include this software in this distribution under the [CDDL or GPL
 * Version 2] license." If you don't indicate a single choice of license, a
 * recipient has the option to distribute your version of this file under
 * either the CDDL, the GPL Version 2 or to extend the choice of license to
 * its licensees as provided above. However, if you add GPL Version 2 code
 * and therefore, elected the GPL Version 2 license, then the option applies
 * only if the new code is made subject to such option by the copyright
 * holder.
 */

1.230 ion-java 1.0.2
1.230.1 Available under license :
Amazon Ion Java
Copyright 2007-2016 Amazon.com, Inc. or its affiliates. All Rights Reserved.
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

1.231 jakarta-el 3.0.4

1.231.1 Available under license :

# Notices for Jakarta Expression Language

This content is produced and maintained by the Jakarta Expression Language project.

* Project home: https://projects.eclipse.org/projects/ee4j.el

## Trademarks

Jakarta Expression Language is a trademark of the Eclipse Foundation.
All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed source code repository logs.

This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0 which is available at http://www.eclipse.org/legal/epl-2.0. This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License v. 2.0 are satisfied: GNU General Public License, version 2 with the GNU Classpath Exception which is available at https://www.gnu.org/software/classpath/license.html.

SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

The project maintains the following source code repositories:

* https://github.com/eclipse-ee4j/el-ri

Content may contain encryption software. The country in which you are currently may have restrictions on the import, possession, and use, and/or re-export to another country, of encryption software. BEFORE using any encryption software, please check the country's laws, regulations and policies concerning the import, possession, or use, and re-export of encryption software, to see if this is permitted.

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial content
   Distributed under this Agreement, and

b) in the case of each subsequent Contributor:
i) changes to the Program, and
ii) additions to the Program;
where such changes and/or additions to the Program originate from
and are Distributed by that particular Contributor. A Contribution
"originates" from a Contributor if it was added to the Program by
such Contributor itself or anyone acting on such Contributor's behalf.
Contributions do not include changes or additions to the Program that
are not Modified Works.

"Contributor" means any person or entity that Distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which
are necessarily infringed by the use or sale of its Contribution alone
or when combined with the Program.

"Program" means the Contributions Distributed in accordance with this
Agreement.

"Recipient" means anyone who receives the Program under this Agreement
or any Secondary License (as applicable), including Contributors.

"Derivative Works" shall mean any work, whether in Source Code or other
form, that is based on (or derived from) the Program and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship.

"Modified Works" shall mean any work in Source Code or other form that
results from an addition to, deletion from, or modification of the
contents of the Program, including, for purposes of clarity any new file
in Source Code form that contains any contents of the Program. Modified
Works shall not include works that contain only declarations,
interfaces, types, classes, structures, or files of the Program solely
in each case in order to link to, bind by name, or subclass the Program
or Modified Works thereof.

"Distribute" means the acts of a) distributing or b) making available
in any manner that enables the transfer of a copy.

"Source Code" means the form of a Program preferred for making
modifications, including but not limited to software source code,
documentation source, and configuration files.

"Secondary License" means either the GNU General Public License,
Version 2.0, or any later versions of that license, including any
exceptions or additional permissions as identified by the initial
Contributor.

2. GRANT OF RIGHTS
a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

e) Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. REQUIREMENTS

3.1 If a Contributor Distributes the Program in any form, then:

a) the Program must also be made available as Source Code, in
accordance with section 3.2, and the Contributor must accompany
the Program with a statement that the Source Code for the Program
is available under this Agreement, and informs Recipients how to
obtain it in a reasonable manner on or through a medium customarily
used for software exchange; and

b) the Contributor may Distribute the Program under a license
different than this Agreement, provided that such license:
   i) effectively disclaims on behalf of all other Contributors all
      warranties and conditions, express and implied, including
      warranties or conditions of title and non-infringement, and
      implied warranties or conditions of merchantability and fitness
      for a particular purpose;

   ii) effectively excludes on behalf of all other Contributors all
       liability for damages, including direct, indirect, special,
       incidental and consequential damages, such as lost profits;

   iii) does not attempt to limit or alter the recipients’ rights
        in the Source Code under section 3.2; and

   iv) requires any subsequent distribution of the Program by any
       party to be under a license that satisfies the requirements
       of this section 3.

3.2 When the Program is Distributed as Source Code:

a) it must be made available under this Agreement, or if the
   Program (i) is combined with other material in a separate file or
   files made available under a Secondary License, and (ii) the initial
   Contributor attached to the Source Code the notice described in
   Exhibit A of this Agreement, then the Program may be made available
   under the terms of such Secondary Licenses, and

b) a copy of this Agreement must be included with each copy of
   the Program.

3.3 Contributors may not remove or alter any copyright, patent,
trademark, attribution notices, disclaimers of warranty, or limitations
of liability ("notices") contained within the Program from any copy of
the Program which they Distribute, provided that Contributors may add
their own appropriate notices.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities
with respect to end users, business partners and the like. While this
license is intended to facilitate the commercial use of the Program,
the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS
SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient
receives no rights or licenses to the intellectual property of any
Contributor under this Agreement, whether expressly, by implication,
estoppel or otherwise. All rights in the Program not expressly granted
under this Agreement are reserved. Nothing in this Agreement is intended
to be enforceable by any entity that is not a Contributor or Recipient.
No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

"This Source Code may also be made available under the following
Secondary Licenses when the conditions for such availability set forth
in the Eclipse Public License, v. 2.0 are satisfied: {name license(s),
version(s), and exceptions or additional permissions here}.”

Simply including a copy of this Agreement, including this Exhibit A
is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular
file, then You may include the notice in a location (such as a LICENSE
file in a relevant directory) where a recipient would be likely to
look for such a notice.

You may add additional accurate notices of copyright ownership.

---

## The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor
Boston, MA 02110-1335
USA

Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to
share and change it. By contrast, the GNU General Public License is
intended to guarantee your freedom to share and change free software--to
make sure the software is free for all its users. This General Public
License applies to most of the Free Software Foundation's software and
to any other program whose authors commit to using it. (Some other Free
Software Foundation software is covered by the GNU Library General
Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price.
Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".
Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this
License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source
code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the
integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH
YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL
NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN
WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY
AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR
DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL
DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM
(INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED
INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF
THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR
OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest
possible use to the public, the best way to achieve this is to make it
free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to
attach them to the start of each source file to most effectively convey
the exclusion of warranty; and each file should have at least the
"copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful, but
WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program; if not, write to the Free Software
Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1335 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this
when it starts in an interactive mode:
Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type
`show w'. This is free software, and you are welcome to redistribute
it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the
appropriate parts of the General Public License. Of course, the commands
you use may be called something other than `show w' and `show c'; they
could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the program, if
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the
program `Gnomovision' (which makes passes at compilers) written by
James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program
into proprietary programs. If your program is a subroutine library, you
may consider it more useful to permit linking proprietary applications
with the library. If this is what you want to do, use the GNU Library
General Public License instead of this License.

---

## CLASSPATH EXCEPTION

Linking this library statically or dynamically with other modules is
making a combined work based on this library. Thus, the terms and
conditions of the GNU General Public License version 2 cover the whole
combination.

As a special exception, the copyright holders of this library give you
permission to link this library with independent modules to produce an
executable, regardless of the license terms of these independent
modules, and to copy and distribute the resulting executable under
terms of your choice, provided that you also meet, for each linked
independent module, the terms and conditions of the license of that
module. An independent module is a module which is not derived from or
based on this library. If you modify this library, you may extend this
exception to your version of the library, but you are not obligated to
do so. If you do not wish to do so, delete this exception statement
from your version.
1.232 checker-qual 3.8.0

1.232.1 Available under license:
Checker Framework qualifiers
Copyright 2004-present by the Checker Framework developers

MIT License:

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.233 jackson-jaxrs-base 2.13.0

1.233.1 Available under license:
This copy of Jackson JSON processor databind module is licensed under the
Apache (Software) License, version 2.0 ("the License").
See the License for details about distribution rights, and the
specific rights regarding derivate works.

You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

1.234 xz-java 1.5

1.234.1 Available under license:

Licensing of XZ for Java

==================================
All the files in this package have been written by Lasse Collin and/or Igor Pavlov. All these files have been put into the public domain. You can do whatever you want with these files.

This software is provided "as is", without any warranty.

1.235 postgresql 42.2.24

1.235.1 Available under license:
Copyright (c) 1997, PostgreSQL Global Development Group
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.236 netty-codec-memcache 4.1.69.Final

1.236.1 Available under license:
No license file was found, but licenses were detected in source scan.

~ Copyright 2013 The Netty Project
~
~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~
~ https://www.apache.org/licenses/LICENSE
2.0
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.

Found in path(s):
* /opt/cola/permits/1235364855_1638382310.0/0/netty-codec-memcache-4-1-69-final-sources.jar/META-INF/maven/io.netty/netty-codec-memcache/pom.xml
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2013 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1235364855_1638382310.0/0/netty-codec-memcache-4-1-69-final-sources.jar/io/netty/handler/codec/memcache/AbstractMemcacheObjectDecoder.java
* /opt/cola/permits/1235364855_1638382310.0/0/netty-codec-memcache-4-1-69-final-sources.jar/io/netty/handler/codec/memcache/binary/AbstractBinaryMemcacheMessage.java
* /opt/cola/permits/1235364855_1638382310.0/0/netty-codec-memcache-4-1-69-final-sources-jar/io/netty/handler/codec/memcache/AbstractMemcacheObject.java
* /opt/cola/permits/1235364855_1638382310.0/0/netty-codec-memcache-4-1-69-final-sources-jar/io/netty/handler/codec/memcache/binary/AbstractBinaryMemcacheEncoder.java
* /opt/cola/permits/1235364855_1638382310.0/0/netty-codec-memcache-4-1-69-final-sources-jar/io/netty/handler/codec/memcache/binary/package-info.java
* /opt/cola/permits/1235364855_1638382310.0/0/netty-codec-memcache-4-1-69-final-sources-jar/io/netty/handler/codec/memcache/binary/FullBinaryMemcacheResponse.java
* /opt/cola/permits/1235364855_1638382310.0/0/netty-codec-memcache-4-1-69-final-sources-jar/io/netty/handler/codec/memcache/binary/AbstractMemcacheObjectAggregator.java
* /opt/cola/permits/1235364855_1638382310.0/0/netty-codec-memcache-4-1-69-final-sources-jar/io/netty/handler/codec/memcache/binary/DefaultMemcacheContent.java
* /opt/cola/permits/1235364855_1638382310.0/0/netty-codec-memcache-4-1-69-final-sources-jar/io/netty/handler/codec/memcache/binary/MemcacheMessage.java
* /opt/cola/permits/1235364855_1638382310.0/0/netty-codec-memcache-4-1-69-final-sources-jar/io/netty/handler/codec/memcache/binary/DefaultLastMemcacheContent.java
* /opt/cola/permits/1235364855_1638382310.0/0/netty-codec-memcache-4-1-69-final-sources-jar/io/netty/handler/codec/memcache/binary/AbstractMemcacheObject.java
* /opt/cola/permits/1235364855_1638382310.0/0/netty-codec-memcache-4-1-69-final-sources-jar/io/netty/handler/codec/memcache/binary/AbstractBinaryMemcacheDecoder.java
* /opt/cola/permits/1235364855_1638382310.0/0/netty-codec-memcache-4-1-69-final-sources-jar/io/netty/handler/codec/memcache/binary/DefaultLastMemcacheContent.java
* /opt/cola/permits/1235364855_1638382310.0/0/netty-codec-memcache-4-1-69-final-sources-jar/io/netty/handler/codec/memcache/binary/FullBinaryMemcacheRequest.java
* /opt/cola/permits/1235364855_1638382310.0/0/netty-codec-memcache-4-1-69-final-sources-jar/io/netty/handler/codec/memcache/binary/AbstractMemcacheObjectAggregator.java
* /opt/cola/permits/1235364855_1638382310.0/0/netty-codec-memcache-4-1-69-final-sources-jar/io/netty/handler/codec/memcache/binary/DefaultMemcacheContent.java
* /opt/cola/permits/1235364855_1638382310.0/0/netty-codec-memcache-4-1-69-final-sources-jar/io/netty/handler/codec/memcache/binary/FullMemcacheMessage.java
* /opt/cola/permits/1235364855_1638382310.0/0/netty-codec-memcache-4-1-69-final-sources-jar/io/netty/handler/codec/memcache/package-info.java

Open Source Used In security_automation identity-guest-cs 2332
1.237 jaxb-core 2.3.0

1.238 spring-integration-kafka 5.4.11

1.238.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2019-2020 the original author or authors.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *   https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
/*
 * Copyright 2018-2020 the original author or authors.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *   https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
Found in path(s):
* /opt/cola/permits/1265673578_1643922079.36/0/spring-integration-kafka-5-4-11-sources-jar/org/springframework/integration/kafka/dsl/KafkaOutboundGatewaySpec.java
* /opt/cola/permits/1265673578_1643922079.36/0/spring-integration-kafka-5-4-11-sources-jar/org/springframework/integration/kafka/inbound/KafkaInboundGateway.java
* /opt/cola/permits/1265673578_1643922079.36/0/spring-integration-kafka-5-4-11-sources-jar/org/springframework/integration/kafka/inbound/KafkaMessageSource.java
* /opt/cola/permits/1265673578_1643922079.36/0/spring-integration-kafka-5-4-11-sources-jar/org/springframework/integration/kafka/dsl/KafkaTemplateSpec.java
* /opt/cola/permits/1265673578_1643922079.36/0/spring-integration-kafka-5-4-11-sources-jar/org/springframework/integration/kafka/dsl/KafkaInboundGatewaySpec.java
* /opt/cola/permits/1265673578_1643922079.36/0/spring-integration-kafka-5-4-11-sources-jar/org/springframework/integration/kafka/dsl/KafkaInboundChannelAdapterSpec.java
* /opt/cola/permits/1265673578_1643922079.36/0/spring-integration-kafka-5-4-11-sources-jar/org/springframework/integration/kafka/support/RawRecordHeaderErrorMessageStrategy.java
* /opt/cola/permits/1265673578_1643922079.36/0/spring-integration-kafka-5-4-11-sources-jar/org/springframework/integration/kafka/support/KafkaSendFailureException.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2017-2019 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *   https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1265673578_1643922079.36/0/spring-integration-kafka-5-4-11-sources-jar/org/springframework/integration/kafka/support/RawRecordHeaderErrorMessageStrategy.java
* /opt/cola/permits/1265673578_1643922079.36/0/spring-integration-kafka-5-4-11-sources-jar/org/springframework/integration/kafka/support/KafkaSendFailureException.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2020 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1265673578_1643922079.36/0/spring-integration-kafka-5-4-11-sources-jar/org/springframework/integration/kafka/dsl/KafkaSubscribableChannelSpec.java
* /opt/cola/permits/1265673578_1643922079.36/0/spring-integration-kafka-5-4-11-sources-jar/org/springframework/integration/kafka/dsl/KafkaPollableChannelSpec.java
* /opt/cola/permits/1265673578_1643922079.36/0/spring-integration-kafka-5-4-11-sources-jar/org/springframework/integration/kafka/support/KafkaIntegrationHeaders.java
* /opt/cola/permits/1265673578_1643922079.36/0/spring-integration-kafka-5-4-11-sources-jar/org/springframework/integration/kafka/channel/PublishSubscribeKafkaChannel.java
* /opt/cola/permits/1265673578_1643922079.36/0/spring-integration-kafka-5-4-11-sources-jar/org/springframework/integration/kafka/dsl/KafkaPointToPointChannelSpec.java
* /opt/cola/permits/1265673578_1643922079.36/0/spring-integration-kafka-5-4-11-sources-jar/org/springframework/integration/kafka/dsl/AbstractKafkaChannelSpec.java
* /opt/cola/permits/1265673578_1643922079.36/0/spring-integration-kafka-5-4-11-sources-jar/org/springframework/integration/kafka/channel/PollableKafkaChannel.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2002-2020 the original author or authors.
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1265673578_1643922079.36/0/spring-integration-kafka-5-4-11-sources-jar/org/springframework/integration/kafka/config/xml/KafkaNamespaceHandler.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2015-2020 the original author or authors.
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1265673578_1643922079.36/0/spring-integration-kafka-5-4-11-sources-jar/org/springframework/integration/kafka/inbound/KafkaMessageDrivenChannelAdapter.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2019 the original author or authors.
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1265673578_1643922079.36/0/spring-integration-kafka-5-4-11-sources-jar/org/springframework/integration/kafka/config/xml/KafkaInboundChannelAdapterParser.java
/*
 * Copyright 2013-2021 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *      https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1265673578_1643922079.36/0/spring-integration-kafka-5-4-11-sources-jar/org/springframework/integration/kafka/config/xml/KafkaInboundGatewayParser.java
* /opt/cola/permits/1265673578_1643922079.36/0/spring-integration-kafka-5-4-11-sources-jar/org/springframework/integration/kafka/config/xml/KafkaOutboundGatewayParser.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2016-2020 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *      https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1265673578_1643922079.36/0/spring-integration-kafka-5-4-11-sources-jar/org/springframework/integration/kafka/outbound/KafkaProducerMessageHandler.java
* /opt/cola/permits/1265673578_1643922079.36/0/spring-integration-kafka-5-4-11-sources-jar/org/springframework/integration/kafka/dsl/Kafka.java
* /opt/cola/permits/1265673578_1643922079.36/0/spring-integration-kafka-5-4-11-sources-jar/org/springframework/integration/kafka/dsl/KafkaProducerMessageHandlerSpec.java

No license file was found, but licenses were detected in source scan.
* /opt/cola/permits/1265673578_1643922079.36/0/spring-integration-kafka-5-4-11-sources.jar/org/springframework/integration/kafka/dsl/KafkaMessageDrivenChannelAdapterSpec.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015-2019 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *     https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1265673578_1643922079.36/0/spring-integration-kafka-5-4-11-sources.jar/org/springframework/integration/kafka/config/xml/KafkaMessageDrivenChannelAdapterParser.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2013-2019 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *     https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1265673578_1643922079.36/0/spring-integration-kafka-5-4-11-sources.jar/org/springframework/integration/kafka/config/xml/KafkaOutboundChannelAdapterParser.java
Apache License
Version 2.0, January 2004
https://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.
Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Apache License
Version 2.0, January 2004
https://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.
"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.
5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
Lettuce Java Redis Client ${version}
Copyright (c) 2011-2020 Mark Paluch

This product is licensed to you under the Apache License, Version 2.0
(the "License"). You may not use this product except in compliance with
the License.

This product may include a number of subcomponents with separate
copyright notices and license terms. Your use of the source code for
these subcomponents is subject to the terms and conditions of the
subcomponent's license, as noted in the license file.

1.240 java-driver-shaded-guava 25.1-jre-graal-sub-1
1.240.1 Available under license:
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2018 The Guava Authors
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
*/

Found in path(s):
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2018 The Guava Authors
 */
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
jar/com/datastax/oss/driver/shaded/guava/common/collect/JdkBackedImmutableSet.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
jar/com/datastax/oss/driver/shaded/guava/common/collect/JdkBackedImmutableMultiset.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
jar/com/datastax/oss/driver/shaded/guava/common/hash/ImmutableSupplier.java

No license file was found, but licenses were detected in source scan.

/*/  
* Copyright (C) 2011 The Guava Authors
* License under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
jar/com/datastax/oss/driver/shaded/guava/common/cache/Cache.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
jar/com/datastax/oss/driver/shaded/guava/common/cache/CacheStats.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
jar/com/datastax/oss/driver/shaded/guava/common/util/concurrent/AsyncFunction.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
jar/com/datastax/oss/driver/shaded/guava/common/util/concurrent/AbstractListeningExecutorService.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
jar/com/datastax/oss/driver/shaded/guava/common/collect/EmptyContiguousSet.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
jar/com/datastax/oss/driver/shaded/guava/common/collect/ImmutableSet.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
jar/com/datastax/oss/driver/shaded/guava/common/collect/ImmutableMultiset.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
jar/com/datastax/oss/driver/shaded/guava/common/collect/ImmutableSupplier.java
No license file was found, but licenses were detected in source scan.

/*
 * Written by Doug Lea with assistance from members of JCP JSR-166
 * Expert Group and released to the public domain, as explained at
 * http://creativecommons.org/publicdomain/zero/1.0/
 */

Found in path(s):
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-jar/com/datastax/oss/driver/shaded/guava/common/hash/Striped64.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-jar/com/datastax/oss/driver/shaded/guava/common/cache/Striped64.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2016 The Guava Authors
 * 
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 * 
 * http://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-jar/com/datastax/oss/driver/shaded/guava/common/base/PatternCompiler.java

No license file was found, but licenses were detected in source scan.
/*
 * Copyright (C) 2005 The Guava Authors
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
jar/com/datastax/oss/driver/shaded/guava/common/reflect/Reflection.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2016 The Guava Authors
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at http://www.apache.org/licenses/LICENSE-2.0
 * Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
jar/com/datastax/oss/driver/shaded/guava/common/collect/ArrayListMultimapGwtSerializationDependencies.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
jar/com/datastax/oss/driver/shaded/guava/common/graph/EndpointPair.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
jar/com/datastax/oss/driver/shaded/guava/common/graph/UndirectedNetworkConnections.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
jar/com/datastax/oss/driver/shaded/guava/common/graph/UndirectedGraphConnections.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
jar/com/datastax/oss/driver/shaded/guava/common/collect/RangeGwtSerializationDependencies.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
jar/com/datastax/oss/driver/shaded/guava/common/graph/ConfigurableMutableNetwork.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
jar/com/datastax/oss/driver/shaded/guava/common/graph/MapIteratorCache.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
jar/com/datastax/oss/driver/shaded/guava/common/graph/ValueGraph.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
jar/com/datastax/oss/driver/shaded/guava/common/graph/ConfigurableNetwork.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
jar/com/datastax/oss/driver/shaded/guava/common/graph/DirectedMultiNetworkConnections.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
jar/com/datastax/oss/driver/shaded/guava/common/graph/ForwardingValueGraph.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
jar/com/datastax/oss/driver/shaded/guava/common/collect/HashMultimapGwtSerializationDependencies.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
jar/com/datastax/oss/driver/shaded/guava/common/graph/ConfigurableNetwork.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
jar/com/datastax/oss/driver/shaded/guava/common/graph/DirectedGraphConnections.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
jar/com/datastax/oss/driver/shaded/guava/common/graph/ElementOrder.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
jar/com/datastax/oss/driver/shaded/guava/common/graph/AbstractDirectedNetworkConnections.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
jar/com/datastax/oss/driver/shaded/guava/common/graph/MutableValueGraph.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
jar/com/datastax/oss/driver/shaded/guava/common/graph/AbstractNetwork.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
jar/com/datastax/oss/driver/shaded/guava/common/graph/DirectedNetworkConnections.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
jar/com/datastax/oss/driver/shaded/guava/common/graph/EndpointPairIterator.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
jar/com/datastax/oss/driver/shaded/guava/common/graph/AbstractGraphBuilder.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2012 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

/*
 * This method was rewritten in Java from an intermediate step of the Murmur hash function in
 * http://code.google.com/p/smhasher/source/browse/trunk/MurmurHash3.cpp, which contained the
MurmurHash3 was written by Austin Appleby, and is placed in the public domain. The author hereby disclaims copyright to this source code.

Copyright (C) 2013 The Guava Authors

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Copyright (C) 2015 The Guava Authors

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-jar/com/datastax/oss/driver/shaded/guava/common/hash/LittleEndianByteArray.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-jar/com/datastax/oss/driver/shaded/guava/common/hash/FarmHashFingerprint64.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2009 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the
* License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND,
either
* express or implied. See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2009 The Guava Authors
jar/com/datastax/oss/driver/shaded/guava/common/collect/CollectSpliterators.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
jar/com/datastax/oss/driver/shaded/guava/common/collect/ImmutableBitMapFauxverideShim.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2006 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 * 
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
jar/com/datastax/oss/driver/shaded/guava/common/util/concurrent/GwtFuturesCatchingSpecialization.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
jar/com/datastax/oss/driver/shaded/guava/common/escape/CharEscaper.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
jar/com/datastax/oss/driver/shaded/guava/common/util/concurrent/Futures.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
jar/com/datastax/oss/driver/shaded/guava/common/util/concurrent/UncheckedTimeoutException.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
jar/com/datastax/oss/driver/shaded/guava/common/util/concurrent/AggregateFuture.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
jar/com/datastax/oss/driver/shaded/guava/common/util/concurrent/ImmediateFuture.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
jar/com/datastax/oss/driver/shaded/guava/common/util/concurrent/FuturesGetChecked.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
jar/com/datastax/oss/driver/shaded/guava/common/util/concurrent/GwtFluentFutureCatchingSpecialization.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
jar/com/datastax/oss/driver/shaded/guava/common/util/concurrent/TimeoutFuture.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
jar/com/datastax/oss/driver/shaded/guava/common/util/concurrent/SimpleTimeLimiter.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
jar/com/datastax/oss/driver/shaded/guava/common/util/concurrent/AbstractTransformFuture.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
jar/com/datastax/oss/driver/shaded/guava/common/util/concurrent/AbstractCatchingFuture.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
jar/com/datastax/oss/driver/shaded/guava/common/util/concurrent/TimeLimiter.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2007 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

/*
 * This following method is a modified version of one found in
 * http://gee.cs.oswego.edu/cgi-bin/viewcvs.cgi/jsr166/src/test/tck/AbstractExecutorServiceTest.java?revision=1.30
 * which contained the following notice:
 *
 * Written by Doug Lea with assistance from members of JCP JSR-166 Expert Group and released to
 * the public domain, as explained at http://creativecommons.org/publicdomain/zero/1.0/
 *
 * Other contributors include Andrew Wright, Jeffrey Hayes, Pat Fisher, Mike Judd.
 */
No license file was found, but licenses were detected in source scan.

*/
* Copyright (C) 2017 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express  
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):

No license file was found, but licenses were detected in source scan.

*/
* Copyright (C) 2007 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express  
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
Open Source Used In security_automation identity-guest-cs 2362
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources.jar/com/datastax/oss/driver/shaded/guava/common/base/Functions.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources.jar/com/datastax/oss/driver/shaded/guava/common/io/ByteStreams.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources.jar/com/datastax/oss/driver/shaded/guava/common/collect/HashBiMap.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources.jar/com/datastax/oss/driver/shaded/guava/common/io/Resources.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources.jar/com/datastax/oss/driver/shaded/guava/common/collect/EnumMultiset.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources.jar/com/datastax/oss/driver/shaded/guava/common/base/Predicates.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources.jar/com/datastax/oss/driver/shaded/guava/common/base/AbstractIterator.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources.jar/com/datastax/oss/driver/shaded/guava/common/base/LineBuffer.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources.jar/com/datastax/oss/driver/shaded/guava/common/eventbus/DeadEvent.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources.jar/com/datastax/oss/driver/shaded/guava/common/base/Throwables.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources.jar/com/datastax/oss/driver/shaded/guava/common/base/Function.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources.jar/com/datastax/oss/driver/shaded/guava/common/io/LittleEndianDataInputStream.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2008 The Guava Authors
*/
Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License
is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
express
or implied. See the License for the specific language governing permissions and limitations under
the License.

Found in path(s):
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-jar/com/datastax/oss/driver/shaded/guava/common/net/PercentEscaper.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-jar/com/datastax/oss/driver/shaded/guava/common/primitives/Longs.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2015 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you
 * may not use this file except in compliance with the License. You may
 * obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
 * implied. See the License for the specific language governing
 * permissions and limitations under the License.
 */

Found in path(s):
/*
 * Copyright (C) 2007 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 */
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
  jar/com/datastax/oss/driver/shaded/guava/common/collection/Multisets.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
  jar/com/datastax/oss/driver/shaded/guava/common/collection/ArrayListMultimap.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
  jar/com/datastax/oss/driver/shaded/guava/common/collection/EnumBiMap.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
  jar/com/datastax/oss/driver/shaded/guava/common/collection/Ordering.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
  jar/com/datastax/oss/driver/shaded/guava/common/collection/ForwardingMapEntry.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
  jar/com/datastax/oss/driver/shaded/guava/common/collection/LexicographicalOrdering.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
  jar/com/datastax/oss/driver/shaded/guava/common/collection/ForwardingMultimap.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
  jar/com/datastax/oss/driver/shaded/guava/common/collection/MapDifference.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
  jar/com/datastax/oss/driver/shaded/guava/common/collection/SetMultimap.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
  jar/com/datastax/oss/driver/shaded/guava/common/collection/AbstractSortedSetMultimap.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
  jar/com/datastax/oss/driver/shaded/guava/common/collection/UsingToStringOrdering.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
  jar/com/datastax/oss/driver/shaded/guava/common/collection/MutableClassToInstanceMap.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
  jar/com/datastax/oss/driver/shaded/guava/common/collection/TreeMultimap.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
  jar/com/datastax/oss/driver/shaded/guava/common/collection/Synchronized.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
  jar/com/datastax/oss/driver/shaded/guava/common/collection/CompoundOrdering.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
  jar/com/datastax/oss/driver/shaded/guava/common/collection/AbstractMapEntry.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
  jar/com/datastax/oss/driver/shaded/guava/common/collection/Lists.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
  jar/com/datastax/oss/driver/shaded/guava/common/collection/TreeMultimap.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources.jar/com/datastax/oss/driver/shaded/guava/common/collection/ForwardingSortedSet.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources.jar/com/datastax/oss/driver/shaded/guava/common/collection/ForwardingCollection.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources.jar/com/datastax/oss/driver/shaded/guava/common/collection/LinkedHashMultimap.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources.jar/com/datastax/oss/driver/shaded/guava/common/collection/ClassToInstanceMap.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources.jar/com/datastax/oss/driver/shaded/guava/common/collection/LinkedHash Multiset.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources.jar/com/datastax/oss/driver/shaded/guava/common/collection/Forwarding Queue.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources.jar/com/datastax/oss/driver/shaded/guava/common/collection/Nulls First Ordering.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources.jar/com/datastax/oss/driver/shaded/guava/common/collection/Forwarding Iterator.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2014 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-jar/com/datastax/oss/driver/shaded/guava/common/graph/Graph.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-jar/com/datastax/oss/driver/shaded/guava/common/graph/ImmutableGraph.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-jar/com/datastax/oss/driver/shaded/guava/common/graph/Network.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-jar/com/datastax/oss/driver/shaded/guava/common/graph/PredecessorsFunction.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-jar/com/datastax/oss/driver/shaded/guava/common/graph/MutableNetwork.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-jar/com/datastax/oss/driver/shaded/guava/common/graph/MutableGraph.java

No license file was found, but licenses were detected in source scan.
* Copyright (C) 2016 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/
/**
/**
* Holder for extra methods of {@code Objects} only in web. Intended to be empty for regular
* version.
*/

Found in path(s):
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
jar/com/datastax/oss/driver/shaded/guava/common/base/ExtraObjectsMethodsForWeb.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2017 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
jar/com/datastax/oss/driver/shaded/guava/common/graph/BaseGraph.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
jar/com/datastax/oss/driver/shaded/guava/common/graph/Traverser.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
jar/com/datastax/oss/driver/shaded/guava/common/graph/AbstractBaseGraph.java
No license file was found, but licenses were detected in source scan.
/*
 * Copyright (C) 2011 The Guava Authors
 * 
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 * http://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2012 The Guava Authors
 * 
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 * 
 * http://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-jar/com/datastax/oss/driver/shaded/guava/common/math/PairedStats.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
/*
* Copyright (C) 2008 The Guava Authors
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

No license file was found, but licenses were detected in source scan.


Open Source Used In security_automation identity-guest-cs 2373
* Copyright (C) 2012 The Guava Authors

* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-jar/com/datastax/oss/driver/shaded/guava/common/collection/UnmodifiableMapKeySet.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-jar/com/datastax/oss/driver/shaded/guava/common/collection/HashBasedTable.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-jar/com/datastax/oss/driver/shaded/guava/thirdparty/publicsuffix/PublicSuffixPatterns.java

No license file was found, but licenses were detected in source scan.
Open Source Used In security_automation

 identity-guest-cs 2376
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2011 The Guava Authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
  jar/com/datastax/oss/driver/shaded/guava/common/hash/package-info.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2010 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
  jar/com/datastax/oss/driver/shaded/guava/common/util/concurrent/ThreadFactoryBuilder.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
  jar/com/datastax/oss/driver/shaded/guava/common/util/concurrent/ListeningExecutorService.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
  jar/com/datastax/oss/driver/shaded/guava/common/base/Strings.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
  jar/com/datastax/oss/driver/shaded/guava/common/primitives/package-info.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
  jar/com/datastax/oss/driver/shaded/guava/common/annotations/Beta.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources.jar/com/datastax/oss/driver/shaded/guava/common/collect/SortedMultiset.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources.jar/com/datastax/oss/driver/shaded/guava/common/collect/ContiguousSet.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources.jar/com/datastax/oss/driver/shaded/guava/common/collect/SortedMultiset.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2011 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not
 * use this file except in compliance with the License. You may obtain a copy of
 * the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources.jar/com/datastax/oss/driver/shaded/guava/common/collect/SortedMultiset.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2008 The Guava Authors
 * 
*/
Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

/*
 * This method was rewritten in Java from an intermediate step of the Murmur hash function in http://code.google.com/p/smhasher/source/browse/trunk/MurmurHash3.cpp, which contained the following header:
 * MurmurHash3 was written by Austin Appleby, and is placed in the public domain. The author hereby disclaims copyright to this source code.
 */

Found in path(s):
*/
Copyright (C) 2009 The Guava Authors

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

*/

Found in path(s):
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2007 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

/**
 * Returns an array containing all of the elements in the specified collection. This method
 * returns the elements in the order they are returned by the collection's iterator. The returned
 * array is "safe" in that no references to it are maintained by the collection. The caller is
 * thus free to modify the returned array.
 */
This method assumes that the collection size doesn't change while the method is running.

TODO(kevinb): support concurrently modified collections?

@param c the collection for which to return an array of elements

*/

Found in path(s):

*/

Copyright (C) 2013 The Guava Authors

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

*/

Found in path(s):

/*
/*
* Copyright (C) 2011 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

/*
* This method was written by Doug Lea with assistance from members of JCP JSR-166 Expert Group
* and released to the public domain, as explained at
* http://creativecommons.org/licenses/publicdomain
*
* As of 2010/06/11, this method is identical to the (package private) hash method in OpenJDK 7's
* java.util.HashMap class.
*/

No license file was found, but licenses were detected in source scan.
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-jar/com/datastax/oss/driver/shaded/guava/common/collection/RowSortedTable.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-jar/com/datastax/oss/driver/shaded/guava/common/collection/MinMaxPriorityQueue.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2014 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2011 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the
 * License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND,
 * either
 * express or implied. See the License for the specific language governing permissions and
 * limitations under the License.
 */

No license file was found, but licenses were detected in source scan.
/*
* Copyright (C) 2009 The Guava Authors
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-jar/com/datastax/oss/driver/shaded/guava/common/coll/SingletonImmutableTable.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-jar/com/datastax/oss/driver/shaded/guava/common/coll/ArrayTable.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-jar/com/datastax/oss/driver/shaded/guava/common/coll/ImmutableTable.java
* /opt/cola/permits/1199743947_1630577692.09/0/java-driver-shaded-guava-25-1-jre-graal-sub-1-sources-
*/
 * Copyright (C) 2009 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */
/**
 * Outer class that exists solely to let us write [@code Partially.GwtIncompatible] instead of plain
 * [@code GwtIncompatible]. This is more accurate for [@link Futures#catching], which is available
 * under GWT but with a slightly different signature.
 *
 * <p>We can't use [@code PartiallyGwtIncompatible] because then the GWT compiler wouldn't recognize
 * it as a [@code GwtIncompatible] annotation. And for [@code Futures.catching], we need the GWT
 * compiler to autostrip the normal server method in order to expose the special, inherited GWT
 * version.
 */

Found in path(s):

1.241 curator-framework 5.1.0

1.241.1 Available under license:

Curator Framework
Copyright 2011-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS.
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at
Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.242 commons-beanutils-core 1.8.3

1.242.1 Available under license:

Apache Commons BeanUtils
Copyright 2000-2010 The Apache Software Foundation

This product includes software developed by
The Apache Software Foundation (http://www.apache.org/).

/*
 * Licensed to the Apache Software Foundation (ASF) under one or more
 * contributor license agreements. See the NOTICE file distributed with
 * this work for additional information regarding copyright ownership.
 * The ASF licenses this file to You under the Apache License, Version 2.0
 * (the "License"); you may not use this file except in compliance with
 * the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.
"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."
"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
1.243 aspect-j 1.9.7
1.243.1 Available under license:
   Found license 'Eclipse Public License 1.0' in "* All rights reserved. This program and the accompanying materials * are made available under the terms of the Eclipse Public License v 2.0"
   Found license 'Eclipse Public License 1.0' in "* All rights reserved. * This program and the accompanying materials are made available * under the terms of the Eclipse Public License v 2.0"

1.244 byte-buddy 1.10.22
1.244.1 Available under license:
   Apache License
   Version 2.0, January 2004
   http://www.apache.org/licenses/

   TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

   1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

   "Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
   Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
   stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
   that You distribute, all copyright, patent, trademark, and
   attribution notices from the Source form of the Work,
   excluding those notices that do not pertain to any part of
   the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of Your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
Copyright ${project.inceptionYear} - ${current.year} ${copyright.holder}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.  
You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software  
distributed under the License is distributed on an "AS IS" BASIS,  
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.  
See the License for the specific language governing permissions and  
limitations under the License.

1.245 netty-codec-socks 4.1.69.Final
1.245.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2013 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1265673557_1643922099.9/0/netty-codec-socks-4-1-69-final-sources-jar/io/netty/handler/codec/socksx/v5/Socks5CommandType.java
* /opt/cola/permits/1265673557_1643922099.9/0/netty-codec-socks-4-1-69-final-sources-jar/io/netty/handler/codec/socks/SocksCmdType.java
* /opt/cola/permits/1265673557_1643922099.9/0/netty-codec-socks-4-1-69-final-sources-jar/io/netty/handler/codec/socks/SocksAddressType.java
* /opt/cola/permits/1265673557_1643922099.9/0/netty-codec-socks-4-1-69-final-sources-jar/io/netty/handler/codec/socks/SocksAuthScheme.java
* /opt/cola/permits/1265673557_1643922099.9/0/netty-codec-socks-4-1-69-final-sources-jar/io/netty/handler/codec/socks/SocksRequestType.java
* /opt/cola/permits/1265673557_1643922099.9/0/netty-codec-socks-4-1-69-final-sources-jar/io/netty/handler/codec/socks/SocksMessageType.java

No license file was found, but licenses were detected in source scan.

~ Copyright 2012 The Netty Project
~
~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~
~ https://www.apache.org/licenses/LICENSE-2.0

~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.

Found in path(s):

No license file was found, but licenses were detected in source scan.

/
* Copyright 2015 The Netty Project
* 
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* 
* https://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
No license file was found, but licenses were detected in source scan.

/*/  
* Copyright 2012 The Netty Project
*  
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*  
* https://www.apache.org/licenses/LICENSE-2.0
*  
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1265673557_1643922099.9/0/netty-codec-socks-4-1-69-final-sources-jar/io/netty/handler/codecsocks/SocksInitRequest.java
* /opt/cola/permits/1265673557_1643922099.9/0/netty-codec-socks-4-1-69-final-sources-jar/io/netty/handler/codecsocks/SocksInitResponse.java
* /opt/cola/permits/1265673557_1643922099.9/0/netty-codec-socks-4-1-69-final-sources-jar/io/netty/handler/codecsoks/SocksRequest.java
* /opt/cola/permits/1265673557_1643922099.9/0/netty-codec-socks-4-1-69-final-sources-jar/io/netty/handler/codecsoks/SocksAuthRequestDecoder.java
* /opt/cola/permits/1265673557_1643922099.9/0/netty-codec-socks-4-1-69-final-sources-jar/io/netty/handler/codecsoks/SocksMessage.java
* /opt/cola/permits/1265673557_1643922099.9/0/netty-codec-socks-4-1-69-final-sources-jar/io/netty/handler/codecsoks/SocksAuthRequestDecoder.java
* /opt/cola/permits/1265673557_1643922099.9/0/netty-codec-socks-4-1-69-final-sources-jar/io/netty/handler/codecsoks/SocksAuthResponse.java
* /opt/cola/permits/1265673557_1643922099.9/0/netty-codec-socks-4-1-69-final-sources-jar/io/netty/handler/codecsoks/SocksCmdRequest.java
* /opt/cola/permits/1265673557_1643922099.9/0/netty-codec-socks-4-1-69-final-sources-jar/io/netty/handler/codecsoks/SocksInitRequestDecoder.java
* /opt/cola/permits/1265673557_1643922099.9/0/netty-codec-socks-4-1-69-final-sources-jar/io/netty/handler/codecsoks/SocksInitResponseDecoder.java
* /opt/cola/permits/1265673557_1643922099.9/0/netty-codec-socks-4-1-69-final-sources-jar/io/netty/handler/codecsoks/SocksCommonUtils.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1265673557_1643922099.9/0/netty-codec-socks-4-1-69-final-sources-jar/io/netty/handler/codec/socks/SocksInitResponseDecoder.java
* /opt/cola/permits/1265673557_1643922099.9/0/netty-codec-socks-4-1-69-final-sources-
1.246 netty-codec-xml 4.1.69.Final

1.246.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014 The Netty Project
 * 
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * 
 *   https://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */
Found in path(s):
* /opt/cola/permits/1235364844_1638382285.3/0/netty-codec-xml-4-1-69-final-sources-jar/io/netty/handler/codec/xml/XmlElementEnd.java
* /opt/cola/permits/1235364844_1638382285.3/0/netty-codec-xml-4-1-69-final-sources-jar/io/netty/handler/codec/xml/XmlComment.java
* /opt/cola/permits/1235364844_1638382285.3/0/netty-codec-xml-4-1-69-final-sources-jar/io/netty/handler/codec/xml/XmlSpace.java
* /opt/cola/permits/1235364844_1638382285.3/0/netty-codec-xml-4-1-69-final-sources-jar/io/netty/handler/codec/xml/XmlCdata.java
* /opt/cola/permits/1235364844_1638382285.3/0/netty-codec-xml-4-1-69-final-sources-jar/io/netty/handler/codec/xml/XmlProcessingInstruction.java
* /opt/cola/permits/1235364844_1638382285.3/0/netty-codec-xml-4-1-69-final-sources-jar/io/netty/handler/codec/xml/XmlContent.java
* /opt/cola/permits/1235364844_1638382285.3/0/netty-codec-xml-4-1-69-final-sources-jar/io/netty/handler/codec/xml/XmlDTD.java

No license file was found, but licenses were detected in source scan.

~ Copyright 2014 The Netty Project
~
~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~
~ https://www.apache.org/licenses/LICENSE-2.0
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.

Found in path(s):

1.247 spring-web 5.3.12

1.247.1 Available under license:
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2021 the original author or authors.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-jar/org/springframework/web/util/pattern/PatternParseException.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-jar/org/springframework/http/codec/json/KotlinSerializationJsonEncoder.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-jar/org/springframework/http/server/reactive/AbstractListenerWriteProcessor.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-jar/org/springframework/http/server/reactive/HttpHandlerDecoratorFactory.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-jar/org/springframework/http/HttpMethod.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-jar/org/springframework/web/bind/MissingRequestHeaderException.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-jar/org/springframework/web/util/pattern/SeparatorPathElement.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-
Open Source Used In security_automation identity-guest-cs 2412

jar/org/springframework/http/codec/support/ClientDefaultCodecsImpl.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-
jar/org/springframework/web/util/pattern/RegexPathElement.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-
jar/org/springframework/web/util/ServletRequestPathUtils.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-
jar/org/springframework/http/codec/HttpMessageWriter.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-
jar/org/springframework/http/converter/json/Jackson2ObjectMapperFactoryBean.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-
jar/org/springframework/web/client/RestTemplate.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-
jar/org/springframework/web/bind/support/WebRequestDataBinder.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-
jar/org/springframework/web/method/support/InvocableHandlerMethod.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-
jar/org/springframework/web/server/adapter/WebHttpHandlerBuilder.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-
jar/org/springframework/http/codec/EncoderHttpMessageWriter.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-
jar/org/springframework/web/context/support/StandardServletEnvironment.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-
jar/org/springframework/http/server/reactive/ServletHttpHandlerAdapter.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-
jar/org/springframework/web/bind/MissingRequestValueException.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-
jar/org/springframework/http/server/reactive/ServerHttpRequestDecorator.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-
jar/org/springframework/web/bind/support/WebExchangeBindException.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-
jar/org/springframework/http/MediaType.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-
jar/org/springframework/web/bind/support/WebExchangeBindException.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-
jar/org/springframework/http/server/reactive/ServerHttpRequestDecorator.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2015 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 */
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-jar/org/springframework/web/context/request/RequestContextListener.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-jar/org/springframework/web/bind/annotation/RequestMethod.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-jar/org/springframework/web/filter/RequestContextFilter.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-jar/org/springframework/web/multipart/MultipartResolver.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-jar/org/springframework/web/server/WebFilterChain.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-jar/org/springframework/http/client/AbstractClientHttpRequestFactoryWrapper.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-jar/org/springframework/http/converter/HttpMessageNotWritableException.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-jar/org/springframework/web/util/HttpSessionMutexListener.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2002-2017 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
Open Source Used In security_automation identity-guest-cs 2418
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2016 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *      https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-jar/org/springframework/http/client/support/HttpRequestWrapper.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-jar/org/springframework/http/ReactiveHttpInputMessage.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-jar/org/springframework/web/bind/annotation/SessionAttribute.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-jar/org/springframework/web/server/session/WebSessionIdResolver.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-jar/org/springframework/web/bind/annotation/PostMapping.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-jar/org/springframework/web/context/request/session/WebSessionManager.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-jar/org/springframework/web/method/annotation/RequestHeaderMapMethodArgumentResolver.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-jar/org/springframework/web/cors/reactive/CorsConfigurationSource.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-jar/org/springframework/web/cors/CorsConfigurationSource.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-jar/org/springframework/web/server/WebHandler.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-jar/org/springframework/web/context/request/SessionScope.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-jar/org/springframework/web/context/annotation/SessionScope.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-jar/org/springframework/web/context/request/RequestContextHolder.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2013 the original author or authors.
 * 
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 *      https://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-jar/org/springframework/web/HttpMediaTypeException.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-jar/org/springframework/http/server/ServerHttpResponse.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2005 the original author or authors.
 * 
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 *      https://www.apache.org/licenses/LICENSE-2.0
 * */
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
*/opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-jar/org/springframework/web/filter/ServletContextRequestLoggingFilter.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2009 the original author or authors.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
*/opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-jar/org/springframework/web/bind/annotation/Mapping.java
No license file was found, but licenses were detected in source scan.

# This file maps Internet media types to unique file extension(s).
# Although created for httpd, this file is used by many software systems
# and has been placed in the public domain for unlimited redistribution.
#
# The table below contains both registered and (common) unregistered types.
# A type that has no unique extension can be ignored -- they are listed
# here to guide configurations toward known types and to make it easier to
# identify "new" types. File extensions are also commonly used to indicate
# content languages and encodings, so choose them carefully.
#
# Internet media types should be registered as described in RFC 4288.
# The registry is at <https://www.iana.org/assignments/media-types/>.
#
# This file was retrieved from
<table>
<thead>
<tr>
<th>MIME type (lowercased)</th>
<th>Extensions</th>
</tr>
</thead>
<tbody>
<tr>
<td>application/1d-interleaved-parityfec</td>
<td></td>
</tr>
<tr>
<td>application/3gpdash-qoe-report+xml</td>
<td></td>
</tr>
<tr>
<td>application/3gpp-ims+xml</td>
<td></td>
</tr>
<tr>
<td>application/a2l</td>
<td></td>
</tr>
<tr>
<td>application/activemessage</td>
<td></td>
</tr>
<tr>
<td>application/alto-costmap+json</td>
<td></td>
</tr>
<tr>
<td>application/alto-costmapfilter+json</td>
<td></td>
</tr>
<tr>
<td>application/alto-directory+json</td>
<td></td>
</tr>
<tr>
<td>application/alto-endpointcost+json</td>
<td></td>
</tr>
<tr>
<td>application/alto-endpointcostparams+json</td>
<td></td>
</tr>
<tr>
<td>application/alto-endpointprop+json</td>
<td></td>
</tr>
<tr>
<td>application/alto-endpointpropparams+json</td>
<td></td>
</tr>
<tr>
<td>application/alto-error+json</td>
<td></td>
</tr>
<tr>
<td>application/alto-networkmap+json</td>
<td></td>
</tr>
<tr>
<td>application/alto-networkmapfilter+json</td>
<td></td>
</tr>
<tr>
<td>application/aml</td>
<td></td>
</tr>
<tr>
<td>application/andrew-insetez</td>
<td></td>
</tr>
<tr>
<td>application/applefile</td>
<td></td>
</tr>
<tr>
<td>application/applixwareaw</td>
<td></td>
</tr>
<tr>
<td>application/atf</td>
<td></td>
</tr>
<tr>
<td>application/atfx</td>
<td></td>
</tr>
<tr>
<td>application/atom+xmlatom</td>
<td></td>
</tr>
<tr>
<td>application/atomcat+xmlatomcat</td>
<td></td>
</tr>
<tr>
<td>application/atomdeleted+xml</td>
<td></td>
</tr>
<tr>
<td>application/atomicmail</td>
<td></td>
</tr>
<tr>
<td>application/atomsvc+xmlatomsvc</td>
<td></td>
</tr>
<tr>
<td>application/atxml</td>
<td></td>
</tr>
<tr>
<td>application/auth-policy+xml</td>
<td></td>
</tr>
<tr>
<td>application/bacnet-xdd+zip</td>
<td></td>
</tr>
<tr>
<td>application/batch-smtp</td>
<td></td>
</tr>
<tr>
<td>application/beep+xml</td>
<td></td>
</tr>
<tr>
<td>application/calendar+json</td>
<td></td>
</tr>
<tr>
<td>application/calendar+xml</td>
<td></td>
</tr>
<tr>
<td>application/call-completion</td>
<td></td>
</tr>
<tr>
<td>application/cals-1840</td>
<td></td>
</tr>
<tr>
<td>application/cbor</td>
<td></td>
</tr>
<tr>
<td>application/ccmp+xml</td>
<td></td>
</tr>
<tr>
<td>application/cjsx+xmlxmlccxml</td>
<td></td>
</tr>
<tr>
<td>application/cdfx+xml</td>
<td></td>
</tr>
<tr>
<td>application/cdmi-capabilitycdmia</td>
<td></td>
</tr>
<tr>
<td>application/cdmi-containercdmic</td>
<td></td>
</tr>
<tr>
<td>application/cdmi-domaincdmid</td>
<td></td>
</tr>
<tr>
<td>application/cdmi-objectcdmio</td>
<td></td>
</tr>
<tr>
<td>application/cdmi-queuecdmiq</td>
<td></td>
</tr>
<tr>
<td>application/each</td>
<td></td>
</tr>
</tbody>
</table>
application/exi
# application/fastinfoset
# application/fastsoap
# application/fdt+xml
# application/fits
# application/font-sfnt
application/font-tdpfr
application/font-woff
# application/framework-attributes+xml
# application/geo+json
application/gml+xml
tgml
application/gpx+xml
gpx
# application/gzip
# application/h224
# application/held+xml
# application/http
application/hyperstudiosk
# application/ibe-key-request+xml
# application/ibe-pkg-reply+xml
# application/ibe-pp-data
# application/iges
# application/im-iscomposing+xml
# application/index
# application/index.cmd
# application/index.obj
# application/index.response
# application/index.vnd
tapplication/inkml+xml
tink
inkml
# application/jot
application/ipfix
# application/ipp
# application/isup
# application/its+xml
application/java-archive
application/java-serialized-object
application/java-vm
application/javascript
# application/jose
# application/jose+json
# application/jrd+json
application/json
# application/json-merge-patch+json
# application/json-seq
application/jsonml+xml
application/jsonml+json
application/jsonml+xml
application/jsonml+json
# application/jwk+json
# application/jwk-set+json
# application/jwt
# application/kpml-request+xml
# application/kpml-response+xml
# application/ld+json
# application/lgr+xml
# application/link-format
# application/load-control+xml
application/lost+xml
# application/lostsync+xml
# application/lxf
application/mac-binhex40
application/mac-compactproct
# application/macwriteii
application/mads+xml
application/marc
application/marcxml+xml
application/mathematica
application/mathml+xml
# application/mathml-content+xml
# application/mathml-presentation+xml
# application/mbms-associated-procedure-description+xml
# application/mbms-deregister+xml
# application/mbms-envelope+xml
# application/mbms-msk+xml
# application/mbms-msk-response+xml
# application/mbms-protection-description+xml
# application/mbms-reception-report+xml
# application/mbms-register+xml
# application/mbms-register-response+xml
# application/mbms-schedule+xml
# application/mbms-user-service-description+xml
application/mbox
# application/media-policy-dataset+xml
# application/media_control+xml
application/mediasearch+xml
# application/merge-patch+json
application/mf4
# application/mikey
application/mets+xml
# application/moss-keys
# application/moss-signature
# application/mosskey-data
# application/mosskey-request
application/mp21
application/mp4
# application/mpeg4-generic
# application/mpeg4-iod
# application/mpeg4-iod-xmt
# application/mrbc-consumer+xml
# application/mrbc-publish+xml
# application/msc-ivr+xml
# application/msc-mixer+xml
application/msword	dot
application/mxf
# application/nasdata
# application/news-checkgroups
# application/news-groupinfo
# application/news-transmission
# application/nlsm+xml
# application/nss
# application/ocsp-request
# application/ocsp-response
application/octet-stream	dms	lrf	mar	sdist	distz	pkg	bapk
dump	ele
deploy
application/oda
# application/odx
application/oebps-package+xml
application/ogx
application/omdoc+xml
application/onenote
application/oxps
# application/p2p-overlay+xml
# application/parityfec
application/patch-ops-error+xml
application/pdf
# application/pdx
application/pgp-encrypted
# application/pgp-keys
application/pgp-signature
application/pics-rules
# application/pidt+xml
# application/pidf-diff+xml
application/pkcs10p10
# application/pkcs12
application/pkcs7-mime7m
application/pkcs7-signature7s
application/pkcs8p8
application/pkix-attr-certac
application/pkix-certcer
application/pkix-crlcrl
application/pkix-pkipatkkipath
application/pkixcmppki
application/pls+xml
application/poc-settings+xml
application/postscript ai eps ps
# application/ppsp-tracker+json
# application/problem+json
# application/problem+xml
# application/provenance+xml
# application/prs.alvestrand.titrax-sheet
application/prs.cwwcww
# application/prs.hpub+zip
# application/prs.nprend
# application/prs.plunker
# application/prs.rdf-xml-crypt
# application/prs.xsf+xml
application/pskc+xml
# application/pskcxml
# application/qsig
# application/raptorfec
# application/rdap+json
application/rdf+xml
application/reginfo+xml
# application/remote-printing
# application/reputon+json
application/resource-lists+xml
application/resource-lists-diff+xml
# application/rfc+xml
# application/rfc
# application/riscos
# application/rmi+xml
application/rlmi+xml
application/rpki-ghostbusters+xml
application/rpki-manifest+xml
application/rpki-roa+xml
# application/rpki-updown
application/rsd+xml
application/rss+xml
application/rdf
# application/rtf
# application/rtf
# application/rtfloopback
# application/rttx
# application/samlassertion+xml
# application/samlmetadata+xml
application/sbml+xml
# application/sbmlxml
# application/scaip+xml
# application/scim+xml
application/scvp-cv-request+xml
application/scvp-cv-response+xml
application/scvp-vp-request+xml
application/scvp-vp-response+xml
application/sdp+xml
# application/sep+xml
# application/sep-exi
# application/ucr-targetdesc+xml
# application/ucr-uisocketdesc+xml
# application/vcard+json
# application/vcard+xml
# application/vemmi
# application/vvidence.scriptfile
# application/vnd.3gpp-prose+xml
# application/vnd.3gpp-prose-pc3ch+xml
# application/vnd.3gpp.access-transfer-events+xml
# application/vnd.3gpp.bsf+xml
# application/vnd.3gpp.mid-call+xml
application/vnd.3gpp.pic-bw-largeplb
application/vnd.3gpp.pic-bw-smallpsb
application/vnd.3gpp.pic-bw-varp
# application/vnd.3gpp.sms
# application/vnd.3gpp.srvcc-ext+xml
# application/vnd.3gpp.srvcc-info+xml
# application/vnd.3gpp.state-and-event-info+xml
# application/vnd.3gpp.usd+xml
# application/vnd.3gpp2.bcmcsinfo+xml
# application/vnd.3gpp2.sms
application/vnd.3gpp2.tcapcap
# application/vnd.3lightssoftware.imagescal
application/vnd.3m.post-it-notespsw
application/vnd.accpac.simply.aso
application/vnd.accpac.simply.imp
application/vnd.accucobolacu
application/vnd.accucorpatc acutc
application/vnd.adobe.air-application-installer-package+zipair
# application/vnd.adobe.flash.movie
application/vnd.adobe.formscentral.fcdtcfdt
application/vnd.adobe.fxpfxpfp xpf
# application/vnd.adobe.partial-upload
application/vnd.adobe.xdpx+xmlxdp
application/vnd.adobe.xdfxdfdf
# application/vnd.aether.imp
# application/vnd.ah-barcode
application/vnd.ahead.spaceahead
application/vnd.airzip.filesecure.azfazf
application/vnd.airzip.filesecure.azsazs
application/vnd.amazon.ebookazw
# application/vnd.amazon.mobi8-ebook
application/vnd.americandynamics.accacc
application/vnd.amiga.amiami
# application/vnd.amundsen.maze+xml
application/vnd.android.package-archiveapk
# application/vnd.anki
application/vnd.anser-web-certificate-issue-initiation
application/vnd.anser-web-funds-transfer-initiation
application/vnd.antix.game-component
# application/vnd.apache.thrift.binary
# application/vnd.apache.thrift.compact
# application/vnd.apache.thrift.json
# application/vnd.api+json
application/vnd.apple.installer+xml
application/vnd.apple.mpegurl
# application/vnd.arastra.swi
application/vnd.aristanetworks.swi
# application/vnd.artsquare
application/vnd.astraea-Software.iota
application/vnd.audiograph
# application/vnd.autopackage
# application/vnd.avistar+xml
# application/vnd.balsamiq.bmml+xml
# application/vnd.balsamiq.bmpr
# application/vnd.bekitzur-stech+json
# application/vnd.biopax.rdf+xml
application/vnd.blueice.multipass
# application/vnd.bluetooth.ep.oob
# application/vnd.bluetooth.le.oob
application/vnd.bmi
application/vnd.businessobjects
# application/vnd.cab-jscript
# application/vnd.canon-cpdl
# application/vnd.canon-lips
# application/vnd.cendio.thinlinc.clientconf
# application/vnd.century-systems.tcp_stream
application/vnd.chemdraw+xml
# application/vnd.chess-pgn
application/vnd.chipnuts.karaoke-mmd
application/vnd.cinderella
# application/vnd.cirpack.isdn-ext
# application/vnd.citationstyles.style+xml
application/vnd.claymore
application/vnd.cloanto.rp9
application/vnd.cinderellacy
# application/vnd.cirpack.isdn-ext
# application/vnd.collection+json
# application/vnd.collection.doc+json
# application/vnd.collection.next+json
# application/vnd.comicbook+zip
# application/vnd.commerce-battelle
application/vnd.commons pace
application/vnd.iccprofile icm
# application/vnd.ieee.1905
application/vnd.igloader igl
application/vnd.immervision-ivp ivp
application/vnd.immervision-ivu ivu
# application/vnd.ims.imscv1p1
# application/vnd.ims.imscv1p2
# application/vnd.ims.imscv1p3
# application/vnd.ims.lis.v2.result+json
# application/vnd.ims.lti.v2.toolconsumerprofile+json
# application/vnd.ims.lti.v2.toolproxy+json
# application/vnd.ims.lti.v2.toolproxy.id+json
# application/vnd.ims.lti.v2.toolsettings+json
# application/vnd.ims.lti.v2.toolsettings.simple+json
# application/vnd.informedcontrol.rms+xml
# application/vnd.informix-visionary
# application/vnd.infotech.project
# application/vnd.infotech.project+xml
# application/vnd.innopath.wamp.notification
application/vnd.insors.igm igm
application/vnd.intercon.formnet xpx
# application/vnd.intertrust.digibox
# application/vnd.intertrust.ncp
application/vnd.intu.qbo qbo
application/vnd.intu.qfx qfx
# application/vnd.iptc.g2.catalogitem+xml
# application/vnd.iptc.g2.conceptitem+xml
# application/vnd.iptc.g2.knowledgeitem+xml
# application/vnd.iptc.g2.newsitem+xml
# application/vnd.iptc.g2.newsmessage+xml
# application/vnd.iptc.g2.packageitem+xml
# application/vnd.iptc.g2.planningitem+xml
application/vnd.ipunplugged.rcprofile rcprofile
application/vnd.irepository.package+xml lrp
application/vnd.is-xpr xpr
application/vnd.isac.fcs fcs
application/vnd.jam jam
# application/vnd.japannet-directory-service
# application/vnd.japannet-jpnstore-wakeup
# application/vnd.japannet-payment-wakeup
# application/vnd.japannet-registration
# application/vnd.japannet-registration-wakeup
# application/vnd.japannet-setstore-wakeup
# application/vnd.japannet-verification
# application/vnd.japannet-verification-wakeup
application/vnd.jcp.javame.midlet-rms rms
application/vnd.jisp jisp
# application/vnd.mitsubishi.misty-guard.trustweb
application/vnd.mobius.dafdaf
application/vnd.mobius.disdis
application/vnd.mobius.mbkmbk
application/vnd.mobius.mqymqy
application/vnd.mobius.msmsl
application/vnd.mobius.plcplc
application/vnd.mobius.txttxtf
application/vnd.mophun.applicationmpn
application/vnd.mophun.certificatempc
# application/vnd.motorola.flexsuite
# application/vnd.motorola.flexsuite.adsi
# application/vnd.motorola.flexsuite.fis
# application/vnd.motorola.flexsuite.gotap
# application/vnd.motorola.flexsuite.kmr
# application/vnd.motorola.flexsuite.ttc
# application/vnd.motorola.flexsuite.wem
# application/vnd.motorola.iprm
application/vnd.mozilla.xul+xml			xul
# application/vnd.ms-3mfdocument
application/vnd.ms-artgalry			cil
# application/vnd.ms-asf
application/vnd.ms-cab-compressedcab
# application/vnd.ms-color.iccprofile
application/vnd.ms-excel.xls xlm xla xlc xlt xlw
application/vnd.ms-excel.addin.macroenabled.12xlam
application/vnd.ms-excel.sheet.binary.macroenabled.12xlsb
application/vnd.ms-excel.sheet.macroenabled.12xlsm
application/vnd.ms-excel.template.macroenabled.12xltx
application/vnd.ms-fontobject
application/vnd.ms-htmlhelp			chm
application/vnd.ms-ims				ims
application/vnd.ms-lrm				lrm
# application/vnd.ms-office.activex+xml
application/vnd.ms-office.theme+xml
# application/vnd.ms-opentype
# application/vnd.ms-package.obfuscated-opentype
application/vnd.ms-pki.seccatcat
application/vnd.ms-pki.stl
# application/vnd.ms-playready.initiator+xml
application/vnd.ms-powerpoint.ppt pps pot
application/vnd.ms-powerpoint.addin.macroenabled.12ppam
application/vnd.ms-powerpoint.presentation.macroenabled.12pptm
application/vnd.ms-powerpoint.slide.macroenabled.12slidm
application/vnd.ms-powerpoint.slideshow.macroenabled.12ppsm
application/vnd.ms-powerpoint.template.macroenabled.12potm
# application/vnd.ms-printdevicecapabilities+xml
# application/vnd.ms-printing.printticket+xml
# application/vnd.openxmlformats-officedocument.themeoverride+xml
# application/vnd.openxmlformats-officedocument.vmldrawing
# application/vnd.openxmlformats-officedocumentwoordprocessingml.comments+xml
application/vnd.openxmlformats-officedocument.wordprocessingml.document
docx
# application/vnd.openxmlformats-officedocument.wordprocessingml.document.glossary+xml
# application/vnd.openxmlformats-officedocument.wordprocessingml.document.main+xml
# application/vnd.openxmlformats-officedocument.wordprocessingml.document.endnotes+xml
# application/vnd.openxmlformats-officedocument.wordprocessingml.fonttable+xml
# application/vnd.openxmlformats-officedocument.wordprocessingml.footer+xml
# application/vnd.openxmlformats-officedocument.wordprocessingml.footnotes+xml
# application/vnd.openxmlformats-officedocument.wordprocessingml.numbering+xml
# application/vnd.openxmlformats-officedocument.wordprocessingml.settings+xml
# application/vnd.openxmlformats-officedocument.wordprocessingml.styles+xml
application/vnd.openxmlformats-officedocument.wordprocessingml.template
dotx
# application/vnd.openxmlformats-officedocument.wordprocessingml.template.main+xml
# application/vnd.openxmlformats-officedocument.wordprocessingml.websettings+xml
# application/vnd.openxmlformats-package.core-properties+xml
# application/vnd.openxmlformats-package.digital-signature-xmlsignature+xml
# application/vnd.openxmlformats-package.relationships+xml
# application/vnd.oracle.resource+json
# application/vnd.orange.indata
application/vnd.osgeo.mapguide.packagemgp
# application/vnd.osgi.bundle
application/vnd.osgi.dpdp
application/vnd.osgi.subsystemesa
# application/vnd.otps.ct-kip+xml
# application/vnd.objl.countgraph
# application/vnd.pagerduty+json
application/vnd.palm.pdb pqa oprc
# application/vnd.panoply
# application/vnd.pao.xml
application/vnd.pawaafilepaw
# application/vnd.pcos
application/vnd.pg.formatstr
application/vnd.pg.osasli6f
# application/vnd.piaccess.application-licence
application/vnd.picsel
application/vnd.pmi.widget
# application/vnd.poc.group-advertisement+xml
application/vnd.pocketlearnplf
application/vnd.powerbuilder6pbd
# application/vnd.powerbuilder6-s
# application/vnd.powerbuilder7
# application/vnd.powerbuilder7-s
# application/vnd.powerbuilder75
# application/vnd.powerbuilder75-s
# application/vnd.preminet
<table>
<thead>
<tr>
<th>MIME Type</th>
<th>Extension</th>
</tr>
</thead>
<tbody>
<tr>
<td>application/x-pkcs7-certreqrespp7r</td>
<td></td>
</tr>
<tr>
<td>application/x-rar-compressed</td>
<td></td>
</tr>
<tr>
<td>application/x-research-info-systemsris</td>
<td></td>
</tr>
<tr>
<td>application/x-shsh</td>
<td></td>
</tr>
<tr>
<td>application/x-shar</td>
<td></td>
</tr>
<tr>
<td>application/x-shockwave-flashtiff</td>
<td></td>
</tr>
<tr>
<td>application/x-silverlight-appxap</td>
<td></td>
</tr>
<tr>
<td>application/x-sqlsql</td>
<td></td>
</tr>
<tr>
<td>application/x-stufffitsit</td>
<td></td>
</tr>
<tr>
<td>application/x-stufffitsitx</td>
<td></td>
</tr>
<tr>
<td>application/x-subripst</td>
<td></td>
</tr>
<tr>
<td>application/x-sv4cpiosv4cpc</td>
<td></td>
</tr>
<tr>
<td>application/x-t3vm-imaget3</td>
<td></td>
</tr>
<tr>
<td>application/x-tadsgam</td>
<td></td>
</tr>
<tr>
<td>application/x-tartar</td>
<td></td>
</tr>
<tr>
<td>application/x-tcltc1</td>
<td></td>
</tr>
<tr>
<td>application/x-textex</td>
<td></td>
</tr>
<tr>
<td>application/x-ustar</td>
<td></td>
</tr>
<tr>
<td>application/x-wais-sourcesrc</td>
<td></td>
</tr>
<tr>
<td># application/x-www-form-urlencoded</td>
<td></td>
</tr>
<tr>
<td>application/x-x509-ca-certder</td>
<td></td>
</tr>
<tr>
<td>application/x-xfigfig</td>
<td></td>
</tr>
<tr>
<td>application/x-xliff+xml</td>
<td></td>
</tr>
<tr>
<td>application/x-xpinstallxpi</td>
<td></td>
</tr>
<tr>
<td>application/x-xzsz</td>
<td></td>
</tr>
<tr>
<td># application/x-zmachinez1</td>
<td></td>
</tr>
<tr>
<td>z2 z3 z4 z5 z6 z7 z8</td>
<td></td>
</tr>
<tr>
<td># application/x400-bp</td>
<td></td>
</tr>
<tr>
<td># application/xacml+xml</td>
<td></td>
</tr>
<tr>
<td>application/xaml+xml</td>
<td></td>
</tr>
<tr>
<td># application/xcap-att+xml</td>
<td></td>
</tr>
<tr>
<td># application/xcap-caps+xml</td>
<td></td>
</tr>
<tr>
<td>application/xcap-diff+xml</td>
<td></td>
</tr>
<tr>
<td># application/xcap-el+xml</td>
<td></td>
</tr>
<tr>
<td># application/xcap-error+xml</td>
<td></td>
</tr>
<tr>
<td># application/xcap-ns+xml</td>
<td></td>
</tr>
<tr>
<td># application/xcon-conference-info+xml</td>
<td></td>
</tr>
<tr>
<td># application/xcon-conference-info-diff+xml</td>
<td></td>
</tr>
<tr>
<td>application/xenc+xml</td>
<td></td>
</tr>
<tr>
<td>application/xhtml+xml</td>
<td></td>
</tr>
<tr>
<td># application/xhtml-voice+xml</td>
<td></td>
</tr>
<tr>
<td>application/xml</td>
<td></td>
</tr>
<tr>
<td># application/xml-dtd</td>
<td></td>
</tr>
<tr>
<td># application/xml-external-parsed-entity</td>
<td></td>
</tr>
<tr>
<td># application/xml-patch</td>
<td></td>
</tr>
</tbody>
</table>
# application/xmpp+xml
application/xop+xml
application/xproc+xml
application/xslt+xml
application/xspf+xml
application/xv+xml
application/yang
application/yin+xml
application/zip
# application/zlib
# audio/1d-interleaved-parityfec
# audio/32kadpcm
# audio/3gpp
# audio/3gpp2
# audio/ac3
audio/adpcm
# audio/amr
# audio/amr-wb
# audio/amr-wb+
# audio/aptx
# audio/asc
# audio/atrac-advanced-lossless
# audio/atrac-x
# audio/atrac3
audio/basic
# audio/bv16
# audio/bv32
# audio/clearmode
# audio/cn
# audio/dat12
# audio/dls
# audio/dsr-es201108
# audio/dsr-es202050
# audio/dsr-es202211
# audio/dsr-es202212
# audio/dv
# audio/dvi4
# audio/eac3
# audio/encaprtp
# audio/evrc
# audio/evrc-qcp
# audio/evrc0
# audio/evrc1
# audio/evrcb
# audio/evrcb0
# audio/evrcb1
# audio/evrcnw
# audio/evrcnw0
# audio/evrcnw1
# audio/evrcwb
# audio/evrcwb0
# audio/evrcwb1
# audio/evs
# audio/example
# audio/fwdred
# audio/g711-0
# audio/g719
# audio/g722
# audio/g7221
# audio/g723
# audio/g726-16
# audio/g726-24
# audio/g726-32
# audio/g726-40
# audio/g728
# audio/g729
# audio/g7291
# audio/g729d
# audio/g729e
# audio/gsm
# audio/gsm-efr
# audio/gsm-hr-08
# audio/ihbc
# audio/ip-mr_v2.5
# audio/isac
# audio/l16
# audio/l20
# audio/l24
# audio/l8
# audio/lpc
audio/midimid midi kar rmi
# audio/mobile-xmf
audio/mp4m4a mp4a
# audio/mp4a-latm
# audio/mpa
# audio/mpa-robust
audio/mpegmpga mp2 mp2a mp3 m2a m3a
# audio/mpeg4-generic
# audio/musepack
audio/oggoga ogg spx
# audio/opus
# audio/parityfec
# audio/pcma
# audio/pcma-wb
# audio/pcmu
# audio/pcmuc-wb
# audio/prs.sid
# audio/qcelp
# audio/raptorfec
# audio/red
# audio/rtp-enc-aescm128
# audio/rtp-midi
# audio/rtploopback
# audio/rtx
audio/s3ms3m
audio/silksil
# audio/smv
# audio/smv-qcp
# audio/smv0
# audio/sp-midi
# audio/speex
# audio/t140c
# audio/t38
# audio/telephone-event
# audio/tone
# audio/uemclip
# audio/ulpfec
# audio/vdvi
# audio/vmr-wb
# audio/vnd.3gpp.iufp
# audio/vnd.4sb
# audio/vnd.audiokoz
# audio/vnd.celp
# audio/vnd.cisco.nse
# audio/vnd.cmles.radio-events
# audio/vnd.cns.anp1
# audio/vnd.cns.inf1
audio/vnd.dece.audiouva uvva
audio/vnd.digital-windeol
# audio/vnd.dlna.adts
# audio/vnd.dolby.heaac.1
# audio/vnd.dolby.heaac.2
# audio/vnd.dolby.mlp
# audio/vnd.dolby.mps
# audio/vnd.dolby.pl2
# audio/vnd.dolby.pl2x
# audio/vnd.dolby.pl2z
# audio/vnd.dolby.pulse.1
audio/vnd.dradra
audio/vnd.dtsdts
audio/vnd.dts.hddtsd
# audio/vnd.dvb.file
# audio/vnd.everad.plj
# audio/vnd.hns.audio
image/x-freehand
fh fh4 fh5 fh7
image/x-iconico
image/x-mrsid-imagesid
image/x-pcxpcx
image/x-pictipcxt
image/x-portable-anymappnm
image/x-portable-bitmappbm
image/x-portable-graymappgm
image/x-portable-pixmapppm
image/x-rbrgb
image/x-tgatga
image/x-xbitmapxbm
image/x-xpixmapxpm
image/x-xwindowdumpxwd
# message/cpim
# message/delivery-status
# message/disposition-notification
# message/example
# message/external-body
# message/feedback-report
# message/global
# message/global-delivery-status
# message/global-disposition-notification
# message/global-headers
# message/http
# message/imdn+xml
# message/news
# message/partial
message/rfc822eml mime
# message/s-http
# message/sip
# message/sipfrag
# message/tracking-status
# message/vnd.si.simp
# message/vnd.wfa.wsc
# model/example
# model/gltf+json
model/igesigs iges
model/mesh mesh silo
model/vnd.collada+xmlmldae
model/vnd.dwf/dwf
# model/vnd.flatland.3dml
model/vnd.gdl/gdl
# model/vnd.gs-gdl
# model/vnd.gs.gdl
model/vnd.gtw/gtw
# model/vnd.moml+xmlml
model/vnd.mtsmts
# model/vnd.opengex
# model/vnd.parasolid.transmit.binary
# model/vnd.parasolid.transmit.text
# model/vnd.rosette.annotated-data-model
# model/vnd.valve.sourcecompiled-map
model/vnd.vtu
model/vrml
model/x3d+binary
# model/x3d+fastinfoset
model/x3d+vrml
model/x3d+xml
# model/x3d-vrml
# multipart/alternative
# multipart/appledouble
# multipart/byteranges
# multipart/digest
# multipart/encrypted
# multipart/example
# multipart/form-data
# multipart/header-set
# multipart/mixed
# multipart/parallel
# multipart/related
# multipart/report
# multipart/signed
# multipart/voice-message
# multipart/x-mixed-replace
# text/1d-interleaved-parityfec
text/cache-manifest
# text/parameters

text/cache-manifest
# text/css
# text/csv
# text/csv-schema
# text/directory
# text/dns
# text/ecmascript
# text/encaprtp
# text/enriched
# text/example
# text/fwdred
# text/grammar-ref-list
text/html
text/javascript
# text/jcr-cnd
# text/markdown
# text/mizar
text/n3
# text/parameters
# video/nv
video/ogg
# video/parityfec
# video:pointer
video/quicktimeqt mov
# video/raptorfec
# video/raw
# video/rtp-enc-aescm128
# video/rtloopback
# video/rtx
# video/smpte292m
# video/ulpfec
# video/vc1
# video/vnd.cctv
video/vnd.dece.hdv uvh
video/vnd.dece.mobi uvm
# video/vnd.dece.mp4
video/vnd.dece.pdv uvp
video/vnd.dece.sdv uvs
video/vnd.dece.videovu uvv
# video/vnd.directv.mpeg
# video/vnd.directv.mpeg-tts
# video/vnd.dlna.mpeg-tts
video/vnd.dvb.file dvb
video/vnd.fvt
# video/vnd.hns.video
# video/vnd.iptvforum.1dparityfec-1010
# video/vnd.iptvforum.1dparityfec-2005
# video/vnd.iptvforum.2dparityfec-1010
# video/vnd.iptvforum.2dparityfec-2005
# video/vnd.iptvforum.tsavc
# video/vnd.iptvforum.tsmpeg2
# video/vnd.motorola.video
# video/vnd.motorola.videop
video/vnd.mpegurlmxu m4u
video/vnd.ms-playready.media.pyv
# video/vnd.nokia.interleaved-multimedia
# video/vnd.nokia.videovoip
# video/vnd.objectvideo
# video/vnd.radgametools.bink
# video/vnd.radgametools.smacker
# video/vnd.sealed.mpeg1
# video/vnd.sealed.mpeg4
# video/vnd.sealed.swf
# video/vnd.sealedmedia.softseal.mov
video/vnd.uvvu.mp4 uvu
video/vnd.vivo
# video/vp8
Open Source Used In security_automation identity-guest-cs  2461

video/webm
video/x-f4v
video/x-flif
video/x-flv
video/x-m4v
video/x-matroska mk3d mks
video/x-mng
video/x-ms-asf asx
video/x-ms-vob
video/x-ms-wm
video/x-ms-wmv
video/x-ms-wmx
video/x-ms-wvx
video/x-sgi-movie movie
video/x-smv
x-conference/x-cooltalk ice

Found in path(s):
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-jar/org/springframework/http/mime.types

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2012 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-jar/org/springframework/web/jsf/DecoratingNavigationHandler.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-jar/org/springframework/web/context/ConfigurableWebEnvironment.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-jar/org/springframework/web/HttpMediaTypeNotAcceptableException.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-jar/org/springframework/web/bind/annotation/ValueConstants.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2010 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *      https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-jar/org/springframework/web/context/support/ServletContextResourceLoader.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2020 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* https://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-
jar/org/springframework/http/converter/xml/MappingJackson2XmlHttpMessageConverter.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-
jar/org/springframework/http/converter/smile/MappingJackson2SmileHttpMessageConverter.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-
jar/org/springframework/web/multipart/support/RequestPartServletServerHttpRequest.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-
jar/org/springframework/web/cors/reactive/DefaultCorsProcessor.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-
jar/org/springframework/web/cors/reactive/CorsUtils.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-
jar/org/springframework/web/context/request/WebRequest.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-
jar/org/springframework/web/context/async/DeferredResult.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-
jar/org/springframework/web/server/handler/ExceptionHandlingWebHandler.java

Open Source Used In security_automation identity-guest-cs 2463
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources.jar/org/springframework/http/server/reactive/AbstractServerHttpResponse.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources.jar/org/springframework/http/server/RequestPath.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources.jar/org/springframework/web/multipart/support/MissingServletRequestPartException.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources.jar/org/springframework/http/server/AbstractServerHttpResponse.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources.jar/org/springframework/http/server/reactive/ReactorResourceFactory.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources.jar/org/springframework/web/context/support/LiveBeansViewServlet.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources.jar/org/springframework/web/context/support/LiveBeansViewServlet.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources.jar/org/springframework/web/context/request/async/WebAsyncManager.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources.jar/org/springframework/web/context/request/async/WebAsyncManager.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources.jar/org/springframework/web/cors/DefaultCorsProcessor.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources.jar/org/springframework/web/cors/DefaultCorsProcessor.java
* Copyright 2002-2018 the original author or authors.
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
  *
  * https://www.apache.org/licenses/LICENSE-2.0
  *
  * Unless required by applicable law or agreed to in writing, software
  * distributed under the License is distributed on an "AS IS" BASIS,
  * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
  * See the License for the specific language governing permissions and
  * limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-
  jar/org/springframework/http/codec/json/Jackson2SmileDecoder.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-
  jar/org/springframework/web/context/request/AbstractRequestAttributes.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-
  jar/org/springframework/web/filter/GenericFilterBean.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-
  jar/org/springframework/web/multipart/support/MultipartFilter.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-
  jar/org/springframework/http/converter/ResourceRegionHttpMessageConverter.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-
  jar/org/springframework/web/context/request/ServletWebRequest.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-
  jar/org/springframework/web/context/support/AbstractRefreshableWebApplicationContext.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-
  jar/org/springframework/web/context/request/async/CallableInterceptorChain.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-
  jar/org/springframework/http/server/reactive/ContextPathCompositeHandler.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-
  jar/org/springframework/web/util/pattern/PathElement.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-
  jar/org/springframework/http/client/HttpComponentsClientHttpRequest.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-
  jar/org/springframework/web/bind/annotation/MatrixVariable.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-
  jar/org/springframework/http/server/remote/RemotingRequestExecutor.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-
  jar/org/springframework/web/client/ResponseErrorHandler.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-
jar/org/springframework/web/bind/annotation/RequestParam.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources.jar/org/springframework/web/filter/RelativeRedirectFilter.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources.jar/org/springframework/remoting/httpinvoker/SimpleHttpInvokerServiceExporter.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources.jar/org/springframework/web/util/DefaultUriTemplateHandler.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources.jar/org/springframework/web/context/ContextLoader.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources.jar/org/springframework/web/context/ServletConfigAware.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources.jar/org/springframework/http/client/reactive/JettyResourceFactory.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources.jar/org/springframework/web/util/OpaqueUriComponents.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources.jar/org/springframework/web/util/JavaScriptUtils.java
No license file was found, but licenses were detected in source scan.

# Copyright 2002-2021 the original author or authors.
#
# Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
# You may obtain a copy of the License at
#
#      https://www.apache.org/licenses/LICENSE-2.0
#
# Unless required by applicable law or agreed to in writing, software
# distributed under the License is distributed on an "AS IS" BASIS,
# WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
# See the License for the specific language governing permissions and
# limitations under the License.
Found in path(s):
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-jar/META-INF/services/reactor.blockhound.integration.BlockHoundIntegration
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2014 the original author or authors.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 *   https://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-jar/org/springframework/web/jsf/FacesContextUtils.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-jar/org/springframework/http/client/ClientHttpRequest.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-jar/org/springframework/http/server/ServerHttpAsyncRequestControl.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-jar/org/springframework/web/context/support/ServletConfigPropertySource.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-jar/org/springframework/web/context/support/ServletContextPropertySource.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-jar/org/springframework/web/context/ConfigurableWebApplicationContext.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-jar/org/springframework/web/context/support/HandlerMethodArgumentResolver.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-jar/org/springframework/web/context/ConfigurableWebApplicationContext.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-jar/org/springframework/web/method/support/HandlerMethodArgumentResolver.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-jar/org/springframework/web/converter/feed/AtomFeedHttpMessageConverter.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2019 the original author or authors.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*     https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
/**
* Return the web session for the current request. Always guaranteed to
* return an instance either matching to the session id requested by the
* client, or with a new session id either because the client did not
* specify one or because the underlying session had expired. Use of this
* method does not automatically create a session. See {@link WebSession}
* for more details.
*/

Found in path(s):
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-
  jar/org/springframework/web/server/ServerWebExchange.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2002-2019 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*     https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-
  jar/org/springframework/http/client/SimpleClientHttpResponse.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-
  jar/org/springframework/http/converter/ByteArrayHttpMessageConverter.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-
  jar/org/springframework/http/client/support/HttpAccessor.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-
jar/org/springframework/http/server/ServerHttpRequest.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-
jar/org/springframework/http/HttpRange.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-
jar/org/springframework/web/method/support/AsyncHandlerMethodReturnValueHandler.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-
jar/org/springframework/http/converter/BufferedImageHttpMessageConverter.java
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-
jar/org/springframework/http/codec/ClientCodecConfigurer.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2002-2007 the original author or authors.
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
*  https://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178512_1635319417.38/0/spring-web-5-3-12-sources-
jar/org/springframework/web/bind/support/SessionStatus.java

1.248 spring-kafka 2.7.1

1.248.1 Available under license:
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2019-2021 the original author or authors.
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
*  https://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/core/reactive/ReactiveKafkaProducerTemplate.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/listener/CompositeRecordInterceptor.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/listener/RecordInterceptor.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/listener/ConsumerProperties.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/support/FailedRecordProcessor.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/support/BatchErrorHandler.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/listener/LoggingProducerListener.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2015-2020 the original author or authors.
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
*     https://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/event/KafkaEvent.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/support/ProducerListener.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/listener/BatchErrorHandler.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/support/LoggingProducerListener.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2016-2021 the original author or authors.
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-
ar/jar/org/springframework/kafka/support/converter/MessagingMessageConverter.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-
ar/jar/org/springframework/kafka/listener/MessageListenerContainer.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-
ar/jar/org/springframework/kafka/listener/adapter/DelegatingInvocableHandler.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-
ar/jar/org/springframework/kafka/retrytopic/DestinationTopicResolver.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-
ar/jar/org/springframework/kafka/core/ConsumerFactory.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-
ar/jar/org/springframework/kafka/listener/ContainerProperties.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-
ar/jar/org/springframework/kafka/listener/GenericMessageListener.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-
ar/jar/org/springframework/kafka/listener/KafkaMessageListenerContainer.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-
ar/jar/org/springframework/kafka/listener/AbstractMessageListenerContainer.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-
ar/jar/org/springframework/kafka/listener/ContainerProperties.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-
ar/jar/org/springframework/kafka/listener/KafkaMessageListenerContainer.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-
ar/jar/org/springframework/kafka/listener/GenericMessageListener.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-
ar/jar/org/springframework/kafka/listener/AbstractMessageListenerContainer.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-
ar/jar/org/springframework/kafka/listener/ContainerProperties.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-
ar/jar/org/springframework/kafka/listener/KafkaMessageListenerContainer.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-
/*
 * Copyright 2002-2020 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources.jar/org/springframework/kafka/listener/adapter/BatchMessagingMessageListenerAdapter.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources.jar/org/springframework/kafka/listener/adapter/MessagingMessageListenerAdapter.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources.jar/org/springframework/kafka/support/converter/BatchMessagingMessageConverter.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2019 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 */

Found in path(s):
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources.jar/org/springframework/kafka/core/ProducerFactoryUtils.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources.jar/org/springframework/kafka/config/KafkaListenableContainerFactory.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources.jar/org/springframework/kafka/config/KafkaListenableEndpoint.java

No license file was found, but licenses were detected in source scan.
https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Found in path(s):

*/

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2018-2020 the original author or authors.
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* https://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/event/ContainerStoppedEvent.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/event/ConsumerStoppingEvent.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/event/ConsumerResumedEvent.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/support/serializer/ErrorHandlingDeserializer.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/event/ConsumerPausedEvent.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/support/CompositeProducerListener.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/event/ConsumerStoppedEvent.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/support/serializer/DeserializationException.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2014-2021 the original author or authors.
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* https://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
No license file was found, but licenses were detected in source scan.

*/

* Copyright 2017-2020 the original author or authors.
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
*      https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2018-2021 the original author or authors.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * https://www.apache.org/licenses/LICENSE-2.0
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-
  jar/org/springframework/kafka/retrytopic/ListenerContainerFactoryConfigurer.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-
  jar/org/springframework/kafka/retrytopic/RetryTopicConstants.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-
  jar/org/springframework/kafka/retrytopic/FixedDelayStrategy.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-
  jar/org/springframework/kafka/listener/TimestampedException.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-
  jar/org/springframework/kafka/listener/PartitionPausingBackOffManagerFactory.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-
  jar/org/springframework/kafka/listener/WakingKafkaConsumerTimingAdjuster.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-
  jar/org/springframework/kafka/support/AllowDenyCollectionManager.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-
  jar/org/springframework/kafka/requestreply/RequestReplyFuture.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-
  jar/org/springframework/kafka/listener/DefaultAfterRollbackProcessor.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-
  jar/org/springframework/kafka/listener/KafkaBackoffException.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-
  jar/org/springframework/kafka/support/Suffixer.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-
  jar/org/springframework/kafka/listener/Rema...
/*
 * Copyright 2020-2021 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 *      https://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

No license file was found, but licenses were detected in source scan.

*/

Found in path(s):
* /opt/cola/errors/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/event/ListenerContainerPartitionNoLongerIdleEvent.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015-2021 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *      https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/core/KafkaTemplate.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/support/serializer/JsonDeserializer.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/core/KafkaOperations.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/listener/ConcurrentMessageListenerContainer.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/listener/ErrorHandler.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/listener/adapter/HandlerAdapter.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/listener/AcknowledgingMessageListener.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2021 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* https://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/retrytopic/RetryTopicNamesProviderFactory.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/core/KafkaAdminOperations.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/support/converter/MappingJacksonParameterizedConverter.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/listener/ListenerContainerRegistry.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/listener/RecoveryStrategy.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/listener/ConsumerAwareRecordRecoverer.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/listener/ConsumerAwareRecordInterceptor.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/retrytopic/SuffixingRetryTopicNamesProviderFactory.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014-2020 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
* https://www.apache.org/licenses/LICENSE-2.0

* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/config/AbstractKafkaListenerEndpoint.java

No license file was found, but licenses were detected in source scan.

/**
* Copyright 2017-2019 the original author or authors.
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/listener/ContainerStoppingBatchErrorHandler.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/support/JacksonPresent.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/listener/KafkaListenerErrorHandler.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/core/KafkaResourceHolder.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/security/jaas/KafkaJaasLoginModuleInitializer.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/support/converter/AbstractJavaTypeMapper.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/listener/ConsumerAwareMessageListener.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/listener/DelegatingMessageListener.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2021 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *      https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/listener/adapter/RecordMessagingMessageListenerAdapter.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2019 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *      https://www.apache.org/licenses/LICENSE-2.0
 *
 */

Open Source Used In security_automation identity-guest-cs 2491
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/annotation/KafkaBootstrapConfiguration.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/annotation/KafkaListenerConfigurer.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2016-2020 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/event/ListenerContainerIdleEvent.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/KafkaException.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/annotation/PartitionOffset.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/support/serializer/ParseStringDeserializer.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/core/ProducerFactory.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/listener/GenericErrorHandler.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/support/serializer/ToStringSerializer.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/...
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2017-2021 the original author or authors.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/core/KafkaProducerException.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/retrytopic/DestinationTopicContainer.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/listener/SeekToCurrentErrorHandler.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/listener/ConsumerAwareErrorHandler.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/listener/ConsumerAwareRebalanceListener.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/listener/SeekToCurrentBatchErrorHandler.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/listener/AcknowledgingConsumerAwareMessageListener.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/retrytopic/DefaultDestinationTopicProcessor.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/core/KafkaAdmin.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/listener/ListenerUtils.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2020 the original author or authors.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 */
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/listener/BatchListenerFailedException.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/event/ListenerContainerNoLongerIdleEvent.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/core/ProducerPostProcessor.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/listener/adapter/ConsumerRecordMetadata.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/listener/adapter/BatchToRecordAdapter.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/support/serializer/JsonTypeResolver.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/support/serializer/SerializationUtils.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/listener/DeliveryAttemptAware.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/listener/KafkaExceptionLogLevelAware.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/listener/ListenerInvokingBatchErrorHandler.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/core/KafkaSendCallback.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/streams/KafkaStreamsMicrometerListener.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/listener/adapter/DefaultBatchToRecordAdapter.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/support/micrometer/MicrometerHolder.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/core/MicrometerProducerListener.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/support/serialization/ToStringSerde.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/config/KafkaStreamsInfrastructureCustomizer.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/core/RoutingKafkaTemplate.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/support/serialization/ToFromStringSerde.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/core/RoutingKafkaTemplate.java

/*
 * Copyright 2018-2019 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *      https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
*/opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources.jar/org/springframework/kafka/support/LogIfLevelEnabled.java
*/opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources.jar/org/springframework/kafka/config/CompositeKafkaStreamsCustomizer.java
*/opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources.jar/org/springframework/kafka/event/ConsumerFailedToStartEvent.java
*/opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources.jar/org/springframework/kafka/support/converter/BytesJsonMessageConverter.java
*/opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources.jar/org/springframework/kafka/core/CleanupConfig.java
*/opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources.jar/org/springframework/kafka/listener/adapter/ReplyHeadersConfigurer.java
*/opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources.jar/org/springframework/kafka/config/KafkaStreamsConfiguration.java
*/opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources.jar/org/springframework/kafka/config/KafkaListenerEndpointAdapter.java
*/opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources.jar/org/springframework/kafka/event/ConsumerStartedEvent.java

No license file was found, but licenses were detected in source scan.
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015-2019 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *      https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
/*
 * /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources.jar/org/springframework/kafka/listener/adapter/InvocationResult.java
 * /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources.jar/org/springframework/kafka/config/KafkaStreamsCustomizer.java
 * /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources.jar/org/springframework/kafka/support/converter/ProjectingMessageConverter.java
 * /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources.jar/org/springframework/kafka/support/TransactionSupport.java
 * /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources.jar/org/springframework/kafka/event/ConsumerStartingEvent.java
No license file was found, but licenses were detected in source scan.
*/
* Copyright 2016-2019 the original author or authors.
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
  
  * https://www.apache.org/licenses/LICENSE-2.0

* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/support/converter/StringJsonMessageConverter.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/listener/adapter/AbstractDelegatingMessageListenerAdapter.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/support/KafkaMessageHeaders.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/listener/BatchLoggingErrorHandler.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/listener/adapter/RecordFilterStrategy.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/listener/AbstractRetryingMessageListenerAdapter.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/support/SendResult.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/listener/adapter/RecordFilterStrategy.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/listener/adapter/AbstractRetryingMessageListenerAdapter.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/support/ConsumerSeekAware.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/support/converter/RecordMessageConverter.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/config/KafkaListenerConfigUtils.java
* /opt/cola/permits/1256060472_1642635524.21/0/spring-kafka-2-7-1-sources-jar/org/springframework/kafka/config/KafkaListenerConfigUtils.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2019-2020 the original author or authors.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

1.249 jakarta xml bind api 2.3.3
1.249.1 Available under license:

Copyright (c) 2017, 2018 Oracle and/or its affiliates. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

- Neither the name of the Eclipse Foundation, Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

[[/]]: # " Copyright (c) 2018, 2019 Oracle and/or its affiliates. All rights reserved. "
[[/]]: # " "
[[/]]: # " This program and the accompanying materials are made available under the "
[[/]]: # " terms of the Eclipse Distribution License v. 1.0, which is available at "
[[/]]: # " "
[[/]]: # " SPDX-License-Identifier: BSD-3-Clause "

# Notices for Jakarta XML Binding

This content is produced and maintained by the Jakarta XML Binding project.

* Project home: https://projects.eclipse.org/projects/ee4j.jaxb

## Trademarks

Jakarta XML Binding is a trademark of the Eclipse Foundation.

## Copyright

All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed
source code repository logs.

## Declared Project Licenses

This program and the accompanying materials are made available under the terms of the Eclipse Distribution License v. 1.0 which is available at http://www.eclipse.org/org/documents/edl-v10.php.

SPDX-License-Identifier: BSD-3-Clause

## Source Code

The project maintains the following source code repositories:

* https://github.com/eclipse-ee4j/jaxb-api
* https://github.com/eclipse-ee4j/jaxb-tck

## Third-party Content

This project leverages the following third party content.

### Apache River (3.0.0)

* License: Apache-2.0 AND BSD-3-Clause

### ASM 7 (n/a)

* License: BSD-3-Clause
* Project: https://asm.ow2.io/
* Source: https://repository.ow2.org/nexus/#nexus-search:gav~org.ow2.asm~asm-commons~~~~~kw,versionexpand

### JTHarness (5.0)

* License: (GPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0)
* Project: https://wiki.openjdk.java.net/display/CodeTools/JT+Harness
* Source: http://hg.openjdk.java.net/code-tools/jtharness/

### normalize.css (3.0.2)

* License: MIT

### SigTest (n/a)

* License: GPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

## Cryptography
Content may contain encryption software. The country in which you are currently may have restrictions on the import, possession, and use, and/or re-export to another country, of encryption software. BEFORE using any encryption software, please check the country's laws, regulations and policies concerning the import, possession, or use, and re-export of encryption software, to see if this is permitted.

1.250 httpcore-nio 4.4.14

1.250.1 Available under license:

Apache HttpCore NIO
Copyright 2005-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
   Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
   stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
   that You distribute, all copyright, patent, trademark, and
   attribution notices from the Source form of the Work,
   excluding those notices that do not pertain to any part of
   the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
   distribution, then any Derivative Works that You distribute must
   include a readable copy of the attribution notices contained
   within such NOTICE file, excluding those notices that do not
   pertain to any part of the Derivative Works, in at least one
   of the following places: within a NOTICE text file distributed
   as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.
APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.251 jain-sip-api 1.2.1
1.251.1 Available under license :
No license file was found, but licenses were detected in source scan.

/**
 *
 * Unpublished - rights reserved under the Copyright Laws of the United States.
 * Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
 * Copyright © 2005 BEA Systems, Inc. All rights reserved.
 *
 * Use is subject to license terms.
 *
 * This distribution may include materials developed by third parties.
 *
 * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 *
 * Module Name   : JSIP Specification
 * File Name     : AuthorizationHeader.java
* Author        : Phelim O'Doherty
*
* HISTORY
* Version   Date   Author  Comments
* 1.1     08/10/2002  Phelim O'Doherty
* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
*/

Found in path(s):
* /opt/cola/permits/1256060690_1646678993.07/0/jain-sip-api-1-2-sources-2.jar/sip/header/AuthorizationHeader.java
No license file was found, but licenses were detected in source scan.

/**
* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
* Unpublished - rights reserved under the Copyright Laws of the United States.
* Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
* Copyright © 2005 BEA Systems, Inc. All rights reserved.
* *
* Use is subject to license terms.
* *
* This distribution may include materials developed by third parties.
* *
* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
*
* Module Name   : JSIP Specification
* File Name     : CallIdHeader.java
* Author        : Phelim O'Doherty
*
* HISTORY
* Version   Date   Author  Comments
* 1.1     08/10/2002  Phelim O'Doherty
* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
*/

Found in path(s):
* /opt/cola/permits/1256060690_1646678993.07/0/jain-sip-api-1-2-sources-2.jar/sip/header/CallIdHeader.java
No license file was found, but licenses were detected in source scan.

/**
* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
* Unpublished - rights reserved under the Copyright Laws of the United States.
* Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
* Copyright © 2005 BEA Systems, Inc. All rights reserved.
* *
* Use is subject to license terms.
* *
* This distribution may include materials developed by third parties.
Module Name   : JSIP Specification
File Name     : RouteHeader.java
Author        : Phelim O'Doherty

* HISTORY
* Version Date Author Comments
* 1.1 08/10/2002 Phelim O'Doherty Initial version
* 1.2 07/07/2005 Phelim O'Doherty Deprecated the completed state.

*/

Found in path(s):
* /opt/cola/permits/1256060690_1646678993.07/0/jain-sip-api-1-2-sources-2-jar/sip/RouteHeader.java
No license file was found, but licenses were detected in source scan.

/**
* Unpublished - rights reserved under the Copyright Laws of the United States.
* Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
* Copyright © 2005 BEA Systems, Inc. All rights reserved.
* Use is subject to license terms.
* This distribution may include materials developed by third parties.
*
Found in path(s):
* /opt/cola/permits/1256060690_1646678993.07/0/jain-sip-api-1-2-sources-2-jar/sip/header/RouteHeader.java
No license file was found, but licenses were detected in source scan.

/**
 * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 * Unpublished - rights reserved under the Copyright Laws of the United States.
 * Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
 * Copyright © 2005 BEA Systems, Inc. All rights reserved.
 * *
 * Use is subject to license terms.
 * *
 * This distribution may include materials developed by third parties.
 * *
 * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 * *
 * Module Name : JSIP Specification
 * File Name : ContentLengthHeader.java
 * Author : Phelim O'Doherty
 * *
 * HISTORY
 * Version Date Author Comments
 * 1.1 08/10/2002 Phelim O'Doherty
 * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 */

Found in path(s):
* /opt/cola/permits/1256060690_1646678993.07/0/jain-sip-api-1-2-sources-2-jar/sip/header/ContentLengthHeader.java
No license file was found, but licenses were detected in source scan.

/**
 * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 * Unpublished - rights reserved under the Copyright Laws of the United States.
 * Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
 * Copyright © 2005 BEA Systems, Inc. All rights reserved.
 * *
 * Use is subject to license terms.
 * *
 * This distribution may include materials developed by third parties.
 * *
 * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 * *
 * Module Name : JSIP Specification
 * File Name : OrganizationHeader.java
 * Author : Phelim O'Doherty
 * *
 * HISTORY
 */
* Version  Date  Author  Comments
* 1.1  08/10/2002  Phelim O'Doherty
* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
*/

Found in path(s):
* /opt/cola/permits/1256060690_1646678993.07/0/jain-sip-api-1-2-sources-2-jar/sip/header/OrganizationHeader.java
No license file was found, but licenses were detected in source scan.

/**
* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
* Unpublished - rights reserved under the Copyright Laws of the United States.
* Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
* Copyright © 2005 BEA Systems, Inc. All rights reserved.
* 
* Use is subject to license terms.
* 
* This distribution may include materials developed by third parties.
* 
* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
*
* Module Name : JSIP Specification
* File Name : Response.java
* Author : Phelim O'Doherty
* 
* HISTORY
* Version  Date  Author  Comments
* 1.1  08/10/2002  Phelim O'Doherty  Initial version
* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
*/

Found in path(s):
* /opt/cola/permits/1256060690_1646678993.07/0/jain-sip-api-1-2-sources-2-jar/sip/message/Response.java
No license file was found, but licenses were detected in source scan.

/**
* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
* Unpublished - rights reserved under the Copyright Laws of the United States.
* Copyright 2003 Sun Microsystems, Inc. All rights reserved.
* Copyright 2005 BEA Systems, Inc. All rights reserved.
* 
* Use is subject to license terms.
* 
* This distribution may include materials developed by third parties.
* 
* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
*
This interface represents the messaging entity of a SIP stack and as such is the interface that defines the messaging and transactional component view of the SIP stack. It must be implemented by any object representing a SIP stack compliant to this specification that interacts directly with a proprietary implementation of a SIP stack.

This interface defines the methods that enable any registered application implementing the [@link javax.sip.SipListener] interface to:

<ul>
  <li> Register a [@link javax.sip.SipListener] to the SipProvider. Once the SipListener is registered with the SipProvider it will get notified of Events representing either Request, Response, Timeout messages, Network errors and Transaction termination.
  <li> De-register a [@link javax.sip.SipListener] from the SipProvider. Once a SipListener is de-registered, it will no longer receive any Events from that SipProvider.
  <li> Send [@link javax.sip.message.Request]'s statelessly.
  <li> Send [@link javax.sip.message.Response]'s statelessly.
  <li> Client and Server Transaction creation methods.
  <li> Listening Point manipulation methods.
  <li> New CallIdHeader accessor method.
  <li> SipStack object accessor method.
</ul>

Architecture:

This specification defines a many-to-one relationship between a SipProvider and a SipStack, a one-to-many relationship between a SipProvider and a ListeningPoint and a many-to-one relationship between a SipProvider and a SipListener.

Each SipProvider can be related to zero or more ListeningPoints. However, the SipProvider can have only one ListeningPoint for each transport type.
* For example, a single SipProvider can have one TCP ListeningPoint and one UDP ListeningPoint but cannot have two UDP ListeningPoints. Applications that wish to have multiple ListeningPoints with the same transport must use a separate SipProvider for each such ListeningPoint.

* The ListeningPoints of the SipProvider specify the local address from where request messages will be sent.

* When the application sends a request directly from a SipProvider or from an object that the SipProvider created (Dialog or Transaction) the application might not know in advance the transport that this request will use. The transport is often resolved using a DNS server. In the process of sending the request, the remote transport will be resolved. The provider will then be able to send the request from a suitable ListeningPoint according to the resolved transport. If the resolved transport is UDP, the Provider's UDP ListeningPoint will be used. If the resolved address is TCP, the Provider's TCP ListeningPoint will be used.

* If the application creates a SipProvider with a single ListeningPoint, let's say UDP, it means that UDP requests will be sent from the specific UDP ListeningPoint, and that the application does not care from where TCP requests will be sent. This is left for the SipStack decision.

* Since the transport might not be known in advance, the application might find it difficult to specify the transport in the Via header. The SipProvider is responsible for fixing the Via header transport if needed. If the application set a sent-by identifier to the ListeningPoint that is different then the sent-by parameter in the message, the sent-by parameter of the message will be updated.

* If the application created a provider with zero ListeningPoint, it must have only a single SipProvider per SipStack.

* A SipProvider has the capability to behave transaction statefully, dialog statefully and statelessly. The transaction stateful methods are defined on the ClientTransaction and ServerTransaction respectfully. The transaction stateful method defined specifically for UAC and stateful proxy applications is:

  * <ul>
  * <li> [@link ClientTransaction#sendRequest()]
  * </ul>

* The stateful (transactional) convenience method defined specifically for UAS and stateful proxy applications is:

  * <ul>
  * <li> [@link ServerTransaction#sendResponse(Response)]
  * </ul>

* The dialog stateful methods defined specifically for UAC and stateful
* proxy applications are:
  * <ul>
  * <li> {@link Dialog#sendRequest(ClientTransaction)}
  * <li> {@link Dialog#sendAck(Request)}
  * </ul>
  * <p>
  * The stateless methods (non-transactional) defined on the SipProvider
  * are:
  * <ul>
  * <li> {@link SipProvider#sendResponse(Response)}
  * <li> {@link SipProvider#sendRequest(Request)}
  * </ul>
  * <p>
  * Transaction Model:<br>
  * This specification supports stateful and stateless applications on a per
  * message basis, hence transactional semantics are not mandated for all
  * messages. This specification defines two types of transactions, server
  * transactions and client transactions. A stateless proxy does not contain a
  * client or server transaction, stateless proxies are effectively transparent
  * with respect to transactions.
  *  * <p>
  * Client Transaction:<br>
  * A client transaction exists between a UAC and a UAS specific to Request
  * messages and a server transaction exists between a UAS and a UAC specific
  * to Response messages. A transaction either server or client identifies
  * messages sent between two SIP entities. The purpose of a client transaction
  * is to identify a Request sent by an application that will reliably deliver
  * the Request to a server transaction on
  * the responding SIP entity. The purpose of a server transaction is to
  * identify a Response sent by an application that will reliably deliver the
  * Response to the request initiator.
  *  * <p>
  * Server Transaction:<br>
  * A new server transaction is required for each response that an application
  * decides to respond to statefully, as follows:
  * <ul>
  * <li>Dialog-Creating Requests: A server transaction is not automatically
  * generated by a SipProvider implementation upon receipt of every
  * Dialog-Creating Request i.e. INVITE. Instead the
  * server transaction is set to <code>null</code> in the RequestEvent and
  * the RequestEvent also containing the Request is passed to the application.
  * It is then the responsibility of the application to decide to handle the
  * Dialog-Creating Request statefully or statelessly, using the appropriate
  * send methods on the SipProvider and the ServerTransaction. If a
  * retransmission of the initial Request request is received by the SipProvider
  * the following procedures should be adhered to:
  * <ul>
  * <li>Determine if an existing transaction is already handling this Request.
* If a transaction exists do not pass the Request to the application via a RequestEvent.
* If a transaction doesn't exist pass the retransmitted request to the application as a RequestEvent.

* Non-Dialog-Creating Requests - When the SipProvider receives a Non-Dialog-Creating Request upon which this application has already responded to the Dialog-Creating Request of the same dialogue the server transaction is automatically placed to the RequestEvent and passed up to the application upon which it can respond. Note that the server transaction may be null in a stateful implementation if the incoming request does not match any dialog but must be part of one. That is for requests that must have state but for which the stack cannot find that state, the application can still handle these requests statelessly. The application cannot create a new server transaction for such requests.

* Sending Requests:
  * The client side of the transport layer is responsible for sending the request. The application passes the the Request to the ClientTransaction Dialog or the SipProvider that will send the Request over one of the SipProvider's ListeningPoints. The SipProvider will choose a ListeningPoint that has the same transport as the destination. For example, a Request that is going to be sent over TCP will use a TCP ListeningPoint. See section 18.1.1 of RFC3261.

* Sending Responses:
  * The server side of the transport layer is responsible for sending the responses. The application passes the Response to the ServerTransaction or the SipProvider that will send the Response over one of its ListeningPoints. The response must be sent from the same ListeningPoint on which the request was received.

* Receiving Requests:
  * A SipProvider should be prepared to receive requests on any IP address, port and transport encapsulated in one of its ListeningPoints. When the SipProvider receives a request over any transport, it must examine the value of the "sent-by" parameter in the top Via header. If the host portion of the "sent-by" parameter contains a domain name, or if it contains an IP address that differs from the packet source address, the server must add a "received" parameter to that Via header field value. This parameter must contain the source address from which the packet was received. This is to assist the SipProvider in sending the response, since it must be sent to the source IP address from which the request came. Next, the SipProvider attempts to match the request to a server transaction. If there are any server transactions in existence, the server transport uses the matching procedures of Chapter 17 of
* RFC3261 to attempt to match the response to an existing transaction. If a matching server transaction is found, the request is passed to that transaction, encapsulated into a RequestEvent and fired to the application for processing. If no match is found, the request is passed to the application, which may decide to construct a new server transaction for that request.

* Receiving Responses

Responses are first processed by the transport layer and then passed up to the transaction layer. The transaction layer performs its processing and then passes the response up to the application. When a response is received, the SipProvider examines the top Via header. If the value of the "sent-by" parameter in that header field value does not correspond to a value that the client transport is configured to insert into requests, the response MUST be silently discarded. If there are any client transactions in existence, the client transport uses the matching procedures of Chapter 17 of RFC3261 to attempt to match the response to an existing transaction. If there is a match, the response must be passed to that transaction, encapsulated into a ResponseEvent and fired to the application. Otherwise, the response is stray and must be passed to the application to determine its outcome i.e. a proxy will forward them, while a User Agent will discard.

* @see SipListener
* @see SipStack
* @author BEA Systems, NIST
* @version 1.2
*

Found in path(s):
* /opt/cola/permits/1256060690_1646678993.07/0/jain-sip-api-1-2-sources-2-jar/sip/SipProvider.java

No license file was found, but licenses were detected in source scan.

/**
 * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 * Unpublished - rights reserved under the Copyright Laws of the United States.
 * Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
 * Copyright © 2005 BEA Systems, Inc. All rights reserved.
 * Use is subject to license terms.
 * This distribution may include materials developed by third parties.
 * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 * Module Name : JSIP Specification
* File Name : HeaderFactory.java
* Author : Phelim O'Doherty
*
* HISTORY
* Version Date Author Comments
* 1.1 08/10/2002 Phelim O'Doherty
* 1.2 20/12/2005 Jereon Van Bemmel Added create methods for PUBLISH headers
* 1.2 20/12/2006 Phelim O'Doherty Added new createCseqHeader with long sequence number
*
*unsupported - rights reserved under the Copyright Laws of the United States.
* Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
* Copyright © 2005 BEA Systems, Inc. All rights reserved.
*
* Use is subject to license terms.
*
* This distribution may include materials developed by third parties.
*
*unsupported - rights reserved under the Copyright Laws of the United States.
* Unpublished rights reserved under the Copyright Laws of the United States.
* Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
* Copyright © 2005 BEA Systems, Inc. All rights reserved.
Module Name : JSIP Specification
File Name : ReasonHeader.java
Author : Phelim O'Doherty

HISTORY
Version Date Author Comments
1.1 08/10/2002 Phelim O'Doherty Initial version, optional header to support RFC3326.

Found in path(s):
/opt/cola/permits/1256060690_1646678993.07/0/jain-sip-api-1-2-sources-2-jar/sip/header/ReasonHeader.java
No license file was found, but licenses were detected in source scan.

/**
 * Unpublished - rights reserved under the Copyright Laws of the United States.
 * Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
 * Copyright © 2005 BEA Systems, Inc. All rights reserved.
 * Use is subject to license terms.
 * This distribution may include materials developed by third parties.
 */

Module Name : JSIP Specification
File Name : TransactionTerminatedEvent.java
Author : Phelim O'Doherty

HISTORY
Version Date Author Comments
1.2 03/05/2005 M. Ranganathan New class

Found in path(s):
/opt/cola/permits/1256060690_1646678993.07/0/jain-sip-api-1-2-sources-2-jar/sip/TransactionTerminatedEvent.java
No license file was found, but licenses were detected in source scan.

/**
 * Unpublished - rights reserved under the Copyright Laws of the United States.
No license file was found, but licenses were detected in source scan.

/**
 * Unpublished - rights reserved under the Copyright Laws of the United States.
 * Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
 * Copyright © 2005 BEA Systems, Inc. All rights reserved.
 * Use is subject to license terms.
 * This distribution may include materials developed by third parties.
 * 
 * Module Name : JSIP Specification
 * File Name : Dialog.java
 * Author : Phelim O'Doherty
 * HISTORY
 * Version Date Author Comments
 * 1.1 08/10/2002 Phelim O'Doherty Initial version
 * 1.2 05/04/2005 M. Rangnathan Added Prack Support
 * Phelim O'Doherty Added createAck method
 * M. Ranganathan Clarified retransmission behavior.
 */

Found in path(s):
* /opt/cola/permits/1256060690_1646678993.07/0/jain-sip-api-1-2-sources-2-jar/sip/Dialog.java
No license file was found, but licenses were detected in source scan.

/**
 * Unpublished - rights reserved under the Copyright Laws of the United States.
 * Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
 * Copyright © 2005 BEA Systems, Inc. All rights reserved.
 * Use is subject to license terms.
 * This distribution may include materials developed by third parties.
 * 
 * Module Name : JSIP Specification
 * File Name : ServerHeader.java
 * Author : Phelim O'Doherty
 */
Found in path(s):
* /opt/cola/permits/1256060690_1646678993.07/0/jain-sip-api-1-2-sources-2-jar/sip/header/ServerHeader.java
No license file was found, but licenses were detected in source scan.

/**
* Unpublished - rights reserved under the Copyright Laws of the United States.
* Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
* Copyright © 2005 BEA Systems, Inc. All rights reserved.
* Use is subject to license terms.
* This distribution may include materials developed by third parties.
*/

Module Name : JSIP Specification
File Name : ContactHeader.java
Author : Phelim O'Doherty

Found in path(s):
* /opt/cola/permits/1256060690_1646678993.07/0/jain-sip-api-1-2-sources-2-jar/sip/header/ContactHeader.java
No license file was found, but licenses were detected in source scan.

/**
* Unpublished - rights reserved under the Copyright Laws of the United States.
* Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
* Copyright © 2005 BEA Systems, Inc. All rights reserved.
* Use is subject to license terms.
* This distribution may include materials developed by third parties.
*/
* Module Name : JSIP Specification
* File Name : DialogDoesNotExistException.java
* Author : Phelim O'Doherty

* HISTORY
* Version Date Author Comments
* 1.2 16/06/2005 Phelim O'Doherty Initial version
*/

* Module Name : JSIP Specification
* File Name : AcceptLanguageHeader.java
* Author : Phelim O'Doherty

* HISTORY
* Version Date Author Comments
* 1.1 08/10/2002 Phelim O'Doherty
*/

No license file was found, but licenses were detected in source scan.

/*
* Unpublished - rights reserved under the Copyright Laws of the United States.
* Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
* Copyright © 2005 BEA Systems, Inc. All rights reserved.
* Use is subject to license terms.
* This distribution may include materials developed by third parties.
*/

No license file was found, but licenses were detected in source scan.

/*
* Unpublished - rights reserved under the Copyright Laws of the United States.
* Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
* Copyright © 2005 BEA Systems, Inc. All rights reserved.
*
Use is subject to license terms.

This distribution may include materials developed by third parties.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Module Name : JSIP Specification
File Name : UnsupportedHeader.java
Author : Phelim O'Doherty

HISTORY
Version Date Author Comments
1.1 08/10/2002 Phelim O'Doherty

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Found in path(s):
/opt/cola/permits/1256060690_1646678993.07/0/jain-sip-api-1-2-sources-2-jar/sip/header/UnsupportedHeader.java
No license file was found, but licenses were detected in source scan.

/** *
 Unpublished - rights reserved under the Copyright Laws of the United States.
 Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
 Copyright © 2005 BEA Systems, Inc. All rights reserved.
 * Use is subject to license terms.
 *
 This distribution may include materials developed by third parties.
 *
 ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Module Name : JSIP Specification
File Name : SipFactory.java
Author : Phelim O'Doherty

HISTORY
Version Date Author Comments
1.1 08/10/2002 Phelim O'Doherty
1.2 20/12/2005 Phelim O'Doherty Modified to enable single stack instance if no IP Address config property.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

The SipFactory is a singleton class which applications can use a single access point to obtain proprietary implementations of this specification. As
the SipFactory is a singleton class there will only ever be one instance of
the SipFactory. The single instance of the SipFactory can be obtained using
the {@link SipFactory#getInstance()} method. If an instance of the SipFactory
already exists it will be returned to the application, otherwise a new
instance will be created. A peer implementation object can be obtained from
the SipFactory by invoking the appropriate create method on the SipFactory
e.g. to create a peer SipStack, an application would invoke the
{@link SipFactory#createSipStack(Properties)} method.

Naming Convention

Note that the SipFactory utilises a naming convention defined by this
specification to identify the location of proprietary objects that implement
this specification. The naming convention is defined as follows:
<ul>
<li>The upper-level package structure referred to by the SipFactory
with the attribute <var>pathname</var> can be used to differentiate between
proprietary implementations from different SIP stack vendors. The
<var>pathname</var> used by each SIP vendor must be the domain name
assigned to that vendor in reverse order. For example, the pathmn used by
Sun Microsystems would be <code>com.sun</code>.
</li>
<li>The lower-level package structure and classname of a peer object
is also mandated by this specification. The lowel-level package must be
identical to the package structure defined by this specification and the
classname is mandated to the interface name appended with the
<code>Impl</code> post-fix. For example, the lower-level package structure
and classname of a proprietary implementation of the
<code>javax.sip.SipStack</code> interface must be
<code>javax.sip.SipStackImpl</code>.
</li>
</ul>

Using this naming convention the SipFactory can locate a vendor's
implementation of this specification without requiring an application to
supply a user defined string to each create method in the SipFactory. Instead
an application merely needs to identify the vendors SIP implementation it
would like to use by setting the pathname of that vendors implementation.

It follows that a proprietary implementation of a peer object of this
specification can be located at:
<p>
<code>'pathname'.lower-level package structure and classname'</code>.

For example an application can use the SipFactory to instantiate a NIST
peer SipStack object by setting the pathname to
<code>gov.nist</code> and calling the createSipStack method. The SipFactory
would return a new instance of the SipStack object at the following location:
<code>gov.nist.java.jsip.SipStackImpl.java</code> Because the space of
domain names is managed, this scheme ensures that collisions between two
different vendor's implementations will not happen. For example: a different
* vendor with a domain name 'bea.com' would have their peer SipStack object
* located at <code>com.bea javax.sip.SipStackImpl.java</code>.
* <p>
* <b>Default Namespace:</b>
* This specification defines a default namespace for the SipFactory, this
* namespace is the location of the Reference Implementation. The default
* namespace is <code>gov.nist</code> the author of the Reference
* Implementation, therefore the <var>pathname</var> will have the initial
* value of <code>gov.nist</code> for a new instance of the SipFactory. An
* application must set the <var>pathname</var> of the SipFactory on retrieval
* of a new instance of the factory in order to use a different vendor's SIP
* stack from that of the Reference Implementation. An application can not mix
* different vendor's peer implementation objects.
* 
* @author BEA Systems, NIST
* @version 1.2
*/

Found in path(s):
* /opt/cola/permits/1256060690_1646678993.07/0/jain-sip-api-1-2-sources-2-jar/sip/SipFactory.java
No license file was found, but licenses were detected in source scan.

/**
 * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 * Unpublished - rights reserved under the Copyright Laws of the United States.
 * Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
 * Copyright © 2005 BEA Systems, Inc. All rights reserved.
 * 
 * Use is subject to license terms.
 * 
 * This distribution may include materials developed by third parties.
 * 
 * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 * 
 * Module Name : JSIP Specification
 * File Name : RequireHeader.java
 * Author : Phelim O'Doherty
 * 
 * HISTORY
 * Version Date Author Comments
 * 1.1 08/10/2002 Phelim O'Doherty
* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
*/

Found in path(s):
* /opt/cola/permits/1256060690_1646678993.07/0/jain-sip-api-1-2-sources-2-jar/header/RequireHeader.java
No license file was found, but licenses were detected in source scan.
/**
 * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 * © 2003 Sun Microsystems, Inc. All rights reserved.
 * © 2005 BEA Systems, Inc. All rights reserved.
 * Use is subject to license terms.
 * This distribution may include materials developed by third parties.
 * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 *
 * Module Name : JSIP Specification
 * File Name : PeerUnavailableException.java
 * Author : Phelim O'Doherty
 *
 * HISTORY
 * Version Date Author Comments
 * 1.1 08/10/2002 Phelim O'Doherty
 */

Found in path(s):
* /opt/cola/permits/1256060690_1646678993.07/0/jain-sip-api-1-2-sources-2-jar/sip/PeerUnavailableException.java
No license file was found, but licenses were detected in source scan.

/**
 * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 * © 2003 Sun Microsystems, Inc. All rights reserved.
 * © 2005 BEA Systems, Inc. All rights reserved.
 * Use is subject to license terms.
 * This distribution may include materials developed by third parties.
 * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 *
 * Module Name : JSIP Specification
 * File Name : ContentLanguageHeader.java
 * Author : Phelim O'Doherty
 *
 * HISTORY
 * Version Date Author Comments
 * 1.1 08/10/2002 Phelim O'Doherty Initial version
 */
No license file was found, but licenses were detected in source scan.

/**
 * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 * Unpublished - rights reserved under the Copyright Laws of the United States.
 * Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
 * Copyright © 2005 BEA Systems, Inc. All rights reserved.
 * Use is subject to license terms.
 * This distribution may include materials developed by third parties.
 * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 * Module Name : JSIP Specification
 * File Name : SipURI.java
 * Author : Phelim O'Doherty
 *
 * HISTORY
 * Version Date Author Comments
 * 1.1 08/10/2002 Phelim O'Doherty Initial version
 * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 */

No license file was found, but licenses were detected in source scan.

/**
 * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 * Unpublished - rights reserved under the Copyright Laws of the United States.
 * Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
 * Copyright © 2005 BEA Systems, Inc. All rights reserved.
 * Use is subject to license terms.
 * This distribution may include materials developed by third parties.
 * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 * Module Name : JSIP Specification
 * File Name : Address.java
 * Author : Phelim O'Doherty
 *
 * HISTORY
 */
* Version  Date      Author          Comments
* 1.1  08/10/2002  Phelim O'Doherty  Initial version
* 1.2  18/05/2005  Phelim O'Doherty  Added hashcode method
* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
*/

Found in path(s):
* /opt/cola/permits/1256060690_1646678993.07/0/jain-sip-api-1-2-sources-2-jar/sip/address/Address.java
No license file was found, but licenses were detected in source scan.

/**
* Unpublished - rights reserved under the Copyright Laws of the United States.
* Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
* Copyright © 2005 BEA Systems, Inc. All rights reserved.
*
* Use is subject to license terms.
*
* This distribution may include materials developed by third parties.
*
* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
* Module Name : JSIP Specification
* File Name   : ListeningPoint.java
* Author      : Phelim O'Doherty
*
* HISTORY
* Version  Date      Author          Comments
* 1.1  08/10/2002  Phelim O'Doherty
* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
*/

Found in path(s):
* /opt/cola/permits/1256060690_1646678993.07/0/jain-sip-api-1-2-sources-2-jar/sip/ListeningPoint.java
No license file was found, but licenses were detected in source scan.

/**
* Unpublished - rights reserved under the Copyright Laws of the United States.
* Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
* Copyright © 2005 BEA Systems, Inc. All rights reserved.
*
* Use is subject to license terms.
*
* This distribution may include materials developed by third parties.
*
Module Name  : JSIP Specification
File Name     : Hop.java
Author        : Phelim O'Doherty

HISTORY
Version   Date      Author              Comments
1.1     19/12/2002  Phelim O'Doherty    Initial version
~/~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
*/

Found in path(s):
* /opt/cola/permits/1256060690_1646678993.07/0/jain-sip-api-1-2-sources-2-jar/sip/address/Hop.java
No license file was found, but licenses were detected in source scan.

/**
* Unpublished - rights reserved under the Copyright Laws of the United States.
* Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
* Copyright © 2005 BEA Systems, Inc. All rights reserved.
* Use is subject to license terms.
* This distribution may include materials developed by third parties.
* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
* Module Name  : JSIP Specification
File Name     : ErrorInfoHeader.java
Author        : Phelim O'Doherty

HISTORY
Version   Date      Author              Comments
1.1     08/10/2002  Phelim O'Doherty    Initial version
~/~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
*/

Found in path(s):
* /opt/cola/permits/1256060690_1646678993.07/0/jain-sip-api-1-2-sources-2-jar/sip/header/ErrorInfoHeader.java
No license file was found, but licenses were detected in source scan.

/**
* Unpublished - rights reserved under the Copyright Laws of the United States.
* Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
* Copyright © 2005 BEA Systems, Inc. All rights reserved.
* Use is subject to license terms.
*
* This distribution may include materials developed by third parties.

* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

* Module Name : JSIP Specification
* File Name   : ReplyToHeader.java
* Author      : Phelim O'Doherty

* HISTORY
* Version   Date      Author              Comments
* 1.1     08/10/2002  Phelim O'Doherty

*~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

*/

Found in path(s):
* /opt/cola/permits/1256060690_1646678993.07/0/jain-sip-api-1-2-sources-2-jar/sip/header/ReplyToHeader.java
No license file was found, but licenses were detected in source scan.

/**
* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
* Unpublished - rights reserved under the Copyright Laws of the United States.
* Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
* Copyright © 2005 BEA Systems, Inc. All rights reserved.
* 
* Use is subject to license terms.
* 
* This distribution may include materials developed by third parties.
* 
* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

* Module Name : JSIP Specification
* File Name   : TransactionUnavailableException.java
* Author      : Phelim O'Doherty

* HISTORY
* Version   Date      Author              Comments
* 1.1     01/04/2003  Phelim O'Doherty

*~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

*/

Found in path(s):
* /opt/cola/permits/1256060690_1646678993.07/0/jain-sip-api-1-2-sources-2-jar/sip/TransactionUnavailableException.java
No license file was found, but licenses were detected in source scan.

/**
* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
* Unpublished - rights reserved under the Copyright Laws of the United States.
No license file was found, but licenses were detected in source scan.

/**
 * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 * Unpublished - rights reserved under the Copyright Laws of the United States.
 * Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
 * Copyright © 2005 BEA Systems, Inc. All rights reserved.
 * * Use is subject to license terms.
 * *
 * This distribution may include materials developed by third parties.
 * *
 * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 *
 * Module Name : JSIP Specification
 * File Name : ToHeader.java
 * Author : Phelim O'Doherty
 * *
 * HISTORY
 * Version Date Author Comments
 * 1.1 08/10/2002 Phelim O'Doherty
 * 12/15/2004 M. Ranganathan Added clarification for To header setTag()
 * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 */

Found in path(s):
* /opt/cola/permits/1256060690_164678993.07/0/jain-sip-api-1-2-sources-2-jar/sip/header/ToHeader.java

No license file was found, but licenses were detected in source scan.

/**
 * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 * Unpublished - rights reserved under the Copyright Laws of the United States.
 * Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
 * Copyright © 2005 BEA Systems, Inc. All rights reserved.
 * *
 * Use is subject to license terms.
 * *
 * This distribution may include materials developed by third parties.
 * *
 * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 *
 * Module Name : JSIP Specification
 * File Name : TransportNotSupportedException.java
 * Author : Phelim O'Doherty
 * *
 * HISTORY
 * Version Date Author Comments
 * 1.1 08/10/2002 Phelim O'Doherty Initial version
Found in path(s):
* /opt/cola/permits/1256060690_1646678993.07/0/jain-sip-api-1-2-sources-2-jar/sip/TransportNotSupportedException.java
No license file was found, but licenses were detected in source scan.

/**
 * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 * Unpublished - rights reserved under the Copyright Laws of the United States.
 * Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
 * Copyright © 2005 BEA Systems, Inc. All rights reserved.
 * *
 * Use is subject to license terms.
 * *
 * This distribution may include materials developed by third parties.
 * *
 * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 * Module Name   : JSIP Specification
 * File Name     : WarningHeader.java
 * Author        : Phelim O'Doherty
 * *
 * HISTORY
 * Version      Date   Author   Comments
 * 1.1   08/10/2002  Phelim O'Doherty
 * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 */

Found in path(s):
* /opt/cola/permits/1256060690_1646678993.07/0/jain-sip-api-1-2-sources-2-jar/sip/header/WarningHeader.java
No license file was found, but licenses were detected in source scan.

/**
 * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 * Unpublished - rights reserved under the Copyright Laws of the United States.
 * Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
 * Copyright © 2005 BEA Systems, Inc. All rights reserved.
 * *
 * Use is subject to license terms.
 * *
 * This distribution may include materials developed by third parties.
 * *
 * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 * Module Name   : JSIP Specification
 * File Name     : ContentDispositionHeader.java
* Author        : Phelim O'Doherty
* HISTORY
* Version  Date  Author       Comments
* 1.1     08/10/2002  Phelim O'Doherty  Initial version
*~/

Found in path(s):
* /opt/cola/permits/1256060690_1646678993.07/0/jain-sip-api-1-2-sources-2-
jar/sip/header/ContentDispositionHeader.java
No license file was found, but licenses were detected in source scan.

/****
* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
* Unpublished - rights reserved under the Copyright Laws of the United States.
* Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
* Copyright © 2005 BEA Systems, Inc. All rights reserved.
* 
* Use is subject to license terms.
* 
* This distribution may include materials developed by third parties.
* 
* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
* 
* Module Name   : JSIP Specification
* File Name     : AuthenticationInfoHeader.java
* Author        : Phelim O'Doherty
* 
* HISTORY
* Version  Date  Author       Comments
* 1.1     08/10/2002  Phelim O'Doherty  Initial version
*~/

Found in path(s):
* /opt/cola/permits/1256060690_1646678993.07/0/jain-sip-api-1-2-sources-2-
jar/sip/header/AuthenticationInfoHeader.java
No license file was found, but licenses were detected in source scan.

/****
* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
* Unpublished - rights reserved under the Copyright Laws of the United States.
* Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
* Copyright © 2005 BEA Systems, Inc. All rights reserved.
* 
* Use is subject to license terms.
* This distribution may include materials developed by third parties.

* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

* Module Name : JSIP Specification
* File Name : FromHeader.java
* Author : Phelim O'Doherty

* HISTORY
* Version Date Author Comments
* 1.1 08/10/2002 Phelim O'Doherty

*/

Found in path(s):
* /opt/cola/permits/1256060690_1646678993.07/0/jain-sip-api-1-2-sources-2-jar/sip/header/FromHeader.java
No license file was found, but licenses were detected in source scan.

/**
* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
* Unpublished - rights reserved under the Copyright Laws of the United States.
* Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
* Copyright © 2005 BEA Systems, Inc. All rights reserved.
* Use is subject to license terms.
* This distribution may include materials developed by third parties.

* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
* Module Name : JSIP Specification
* File Name : RetryAfterHeader.java
* Author : Phelim O'Doherty

* HISTORY
* Version Date Author Comments
* 1.1 08/10/2002 Phelim O'Doherty

*/

Found in path(s):
* /opt/cola/permits/1256060690_1646678993.07/0/jain-sip-api-1-2-sources-2-jar/sip/header/RetryAfterHeader.java
No license file was found, but licenses were detected in source scan.

/**
* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
* Unpublished - rights reserved under the Copyright Laws of the United States.
* Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
* Copyright © 2005 BEA Systems, Inc. All rights reserved.
* Use is subject to license terms.
* This distribution may include materials developed by third parties.
* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
* Module Name   : JSIP Specification
* File Name     : AllowEventsHeader.java
* Author        : Phelim O'Doherty
* HISTORY
* Version   Date      Author              Comments
* 1.1     13/12/2002  Phelim O'Doherty    Initial version, extension header to
  support RFC3265
* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
* Found in path(s):
* /opt/cola/permits/1256060690_1646678993.07/0/jain-sip-api-1-2-sources-2-jar/sip/header/AllowEventsHeader.java
No license file was found, but licenses were detected in source scan.

/**
* Unpublished - rights reserved under the Copyright Laws of the United States.
* Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
* Copyright © 2005 BEA Systems, Inc. All rights reserved.
* Use is subject to license terms.
* This distribution may include materials developed by third parties.
* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
* Module Name   : JSIP Specification
* File Name     : ViaHeader.java
* Author        : Phelim O'Doherty
* HISTORY
* Version   Date      Author              Comments
* 1.1     08/10/2002  Phelim O'Doherty
* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
*/

Found in path(s):
* /opt/cola/permits/1256060690_1646678993.07/0/jain-sip-api-1-2-sources-2-jar/sip/header/ViaHeader.java
No license file was found, but licenses were detected in source scan.

/**
 * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 * Unpublished - rights reserved under the Copyright Laws of the United States.
 * Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
 * Copyright © 2005 BEA Systems, Inc. All rights reserved.
 *
 * Use is subject to license terms.
 *
 * This distribution may include materials developed by third parties.
 *
 * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 *
 * Module Name : JSIP Specification
 * File Name   : CSeqHeader.java
 * Author      : Phelim O'Doherty
 *
 * HISTORY
 * Version   Date      Author              Comments
 * 1.1       08/10/2002 Phelim O'Doherty
 * 1.2       11/06/2006 Phelim O'Doherty    Changed Cseq to type long
 *
 * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 */

Found in path(s):
* /opt/cola/permits/1256060690_1646678993.07/0/jain-sip-api-1-2-sources-2-jar/sip/header/CSeqHeader.java

No license file was found, but licenses were detected in source scan.

/**
 * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 * Unpublished - rights reserved under the Copyright Laws of the United States.
 * Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
 * Copyright © 2005 BEA Systems, Inc. All rights reserved.
 *
 * Use is subject to license terms.
 *
 * This distribution may include materials developed by third parties.
 *
 * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 *
 * Module Name : JSIP Specification
 * File Name   : TransactionState.java
 * Author      : Phelim O'Doherty
 *
 * HISTORY
 * Version   Date      Author              Comments
* 1.1 08/10/2002 Phelim O'Doherty Initial version
* 1.2 19/05/2005 Phelim O'Doherty Added equals and hashcode method
* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
*/

Found in path(s):
* /opt/cola/permits/1256060690_1646678993.07/0/jain-sip-api-1-2-sources-2-jar/sip/TransactionState.java
No license file was found, but licenses were detected in source scan.

/**
* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
* Unpublished - rights reserved under the Copyright Laws of the United States.
* Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
* Copyright © 2005 BEA Systems, Inc. All rights reserved.
* Use is subject to license terms.
*
* This distribution may include materials developed by third parties.
* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
* Module Name : JSIP Specification
* File Name : SIPETagHeader.java
* Author : Jeroen van Bemmel
* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
* HISTORY
* Version Date Author Comments
* 1.1 27/10/2005 Jeroen van Bemmel Initial version, header to support RFC3903.
* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
*/

Found in path(s):
* /opt/cola/permits/1256060690_1646678993.07/0/jain-sip-api-1-2-sources-2-jar/header/SIPETagHeader.java
No license file was found, but licenses were detected in source scan.

/**
* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
* Unpublished - rights reserved under the Copyright Laws of the United States.
* Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
* Copyright © 2005 BEA Systems, Inc. All rights reserved.
* Use is subject to license terms.
*
* This distribution may include materials developed by third parties.
* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
* Module Name : JSIP Specification
* File Name : Parameters.java
* Author : Phelim O'Doherty
*
* HISTORY
* Version Date Author Comments
* 1.1 08/10/2002 Phelim O'Doherty Initial version
*~/

Found in path(s):
*/
* /opt/cola/permits/1256060690_1646678993.07/0/jain-sip-api-1-2-sources-2-jar/sip/header/Parameters.java
No license file was found, but licenses were detected in source scan.

/****
* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
* Unpublished - rights reserved under the Copyright Laws of the United States.
* Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
* Copyright © 2005 BEA Systems, Inc. All rights reserved.
* 
* Use is subject to license terms.
* 
* This distribution may include materials developed by third parties.
* 
* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
* 
* Module Name : JSIP Specification
* File Name : ProxyRequireHeader.java
* Author : Phelim O'Doherty
*
* HISTORY
* Version Date Author Comments
* 1.1 08/10/2002 Phelim O'Doherty
*~/

Found in path(s):
* /opt/cola/permits/1256060690_1646678993.07/0/jain-sip-api-1-2-sources-2-jar/sip/header/ProxyRequireHeader.java
No license file was found, but licenses were detected in source scan.

/****
* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
* Unpublished - rights reserved under the Copyright Laws of the United States.
* Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
* Copyright © 2005 BEA Systems, Inc. All rights reserved.
* 
* Use is subject to license terms.
*
* This distribution may include materials developed by third parties.
 *
* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
*
* Module Name   : JSIP Specification
* File Name     : DialogTerminatedEvent.java
* Author        : Phelim O'Doherty
*
* HISTORY
* Version   Date   Author       Comments
* 1.2       17/06/2005  Pnelim O'Doherty New class
*
*~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
*
/*

Found in path(s):
* /opt/cola/permits/1256060690_1646678993.07/0/jain-sip-api-1-2-sources-2-jar/sip/DialogTerminatedEvent.java
No license file was found, but licenses were detected in source scan.

/**
 * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 * Unpublished - rights reserved under the Copyright Laws of the United States.
 * Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
 * Copyright © 2005 BEA Systems, Inc. All rights reserved.
 * *
 * Use is subject to license terms.
 * *
 * This distribution may include materials developed by third parties.
 * *
* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
*
* Module Name   : JSIP Specification
* File Name     : InReplyToHeader.java
* Author        : Phelim O'Doherty
*
* HISTORY
* Version   Date   Author       Comments
* 1.1       08/10/2002  Phelim O'Doherty Initial version
*~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
*
/*

Found in path(s):
* /opt/cola/permits/1256060690_1646678993.07/0/jain-sip-api-1-2-sources-2-jar/header/InReplyToHeader.java
No license file was found, but licenses were detected in source scan.

/**
 * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 * Unpublished - rights reserved under the Copyright Laws of the United States.
1.1 08/10/2002 Phelim O'Doherty
* Unpublished - rights reserved under the Copyright Laws of the United States.
* Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
* Copyright © 2005 BEA Systems, Inc. All rights reserved.
* Use is subject to license terms.
* This distribution may include materials developed by third parties.
*/

No license file was found, but licenses were detected in source scan.

/**
* Unpublished - rights reserved under the Copyright Laws of the United States.
* Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
* Copyright © 2005 BEA Systems, Inc. All rights reserved.
* Use is subject to license terms.
* This distribution may include materials developed by third parties.
*/

No license file was found, but licenses were detected in source scan.

/**
* Unpublished - rights reserved under the Copyright Laws of the United States.
* Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
* Copyright © 2005 BEA Systems, Inc. All rights reserved.
*/

No license file was found, but licenses were detected in source scan.
* Use is subject to license terms.

* This distribution may include materials developed by third parties.

* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

* Module Name : JSIP Specification
* File Name : HeaderAddress.java
* Author : Phelim O'Doherty

* HISTORY
* Version Date Author Comments
* 1.1 08/10/2002 Phelim O'Doherty Initial version
*~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

*/

Found in path(s):
* /opt/cola/permits/1256060690_1646678993.07/0/jain-sip-api-1-2-sources-2-jar/sip/header/HeaderAddress.java
No license file was found, but licenses were detected in source scan.

/**
* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

* Unpublished - rights reserved under the Copyright Laws of the United States.
* Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
* Copyright © 2005 BEA Systems, Inc. All rights reserved.
* 
* Use is subject to license terms.
*
* This distribution may include materials developed by third parties.
*
* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

* Module Name : JSIP Specification
* File Name : Header.java
* Author : Phelim O'Doherty

* HISTORY
* Version Date Author Comments
* 1.1 08/10/2002 Phelim O'Doherty
* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

*/

Found in path(s):
* /opt/cola/permits/1256060690_1646678993.07/0/jain-sip-api-1-2-sources-2-jar/sip/header/Header.java
No license file was found, but licenses were detected in source scan.

/**
* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
This interface represents the management interface of a SIP stack implementing this specification and as such is the interface that defines the management/architectural view of the SIP stack. It defines the methods required to represent and provision a proprietary SIP protocol stack. 

This SipStack interface defines the methods that are be used by an application implementing the `javax.sip.SipListener` interface to control the architecture and setup of the SIP stack. These methods include:

- Creation/deletion of `javax.sip.SipProvider`'s that represent messaging objects that can be used by an application to send `javax.sip.message.Request` and `javax.sip.message.Response` messages statelessly or statefully via Client and Server transactions. 
- Creation/deletion of `javax.sip.ListeningPoint`'s that represent different ports and transports that a SipProvider can use to send and receive messages.

Architecture:

This specification mandates a one-to-many relationship between a SipStack and a SipProvider. There is a one-to-many relationship between a SipStack and a ListeningPoint.
* An application must create a SipStack by invoking the
* `@link SipFactory#createSipStack(Properties)` method, ensuring the
* `@link SipFactory#setPathName(String)` is set. Following the naming
* convention defined in `@link javax.sip.SipFactory`, the implementation of
* the SipStack interface must be called SipStackImpl. This specification also
* defines a stack configuration mechanism using `java.util.Properties`, therefore
* this constructor must also accept a properties argument:
* `<p>`
* `<center>`public SipStackImpl(Properties properties) {} </center>`<p>`
* `</p>`
* The following table documents the static configuration properties which can
* be set for an implementation of a SipStack. This specification doesn’t
* preclude additional values within a configuration properties object if
* understood by the underlying implementation. In order to change these
* properties after a SipStack has been initialized the SipStack must be deleted
* and recreated:
* `<p>`
* `<center>`<table border="1" bordercolorlight="#FFFFFF"
* bordercolordark="#000000" width="98%" cellpadding="3" cellspacing="0">
* `<p class="title">
* `<tr bgcolor="#CCCCCC">
* `<th align="left" valign="top">
* `<p class="table">
* `<strong><strong>SipStack Property </strong> </strong></th>
* `<p class="table">
* `<th align="left" valign="top">Description </th>
* `</tr>`
* `<tr bgcolor="#CCCCCC">
* `<td align="left" valign="top">javax.sip.IP_ADDRESS</td>
* `<p class="table">
* `<b>Deprecated v1.2.</b> It is recommended in this specification that the IP
* Address should be set using the enhanced ListeningPoint architecture,
* therefore this property is no longer mandatory. When this parameter is
* specified as null, a singleton stack instance will be created and returned
* by the SipFactory and the IP Address attributes can be managed via the
* `@link SipStack#createListeningPoint(String, int, String)` method. For
* backwards compatibility if this flag is set the SipFactory will return a new
* SipStack instance each time a SipStack is created with a new IP Address.
* This configuration parameter will become the default IP address of the
* SipStack. The SIP Factory will return any existing instance of SipStack
* that already exist for this IP Address.

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>javax.sip.STACK_NAME</td>
<td>Sets a user friendly name to identify the underlying stack implementation to</td>
</tr>
<tr>
<td></td>
<td>the property value i.e. NISTv1.2. The stack name property should contain no</td>
</tr>
<tr>
<td></td>
<td>spaces. This property is mandatory.</td>
</tr>
<tr>
<td>javax.sip.OUTBOUND_PROXY</td>
<td>Sets the outbound proxy of the SIP Stack. The format for this string is</td>
</tr>
<tr>
<td></td>
<td>ipaddress:port/transport i.e. 129.1.22.333:5060/UDP. This property is</td>
</tr>
<tr>
<td></td>
<td>optional.</td>
</tr>
<tr>
<td>javax.sip.ROUTER_PATH</td>
<td>Sets the fully qualified classpath to the application supplied Router object</td>
</tr>
<tr>
<td></td>
<td>that determines how to route messages when the stack cannot make a routing</td>
</tr>
<tr>
<td></td>
<td>decision (i.e. non-sip URIs). In version 1.2 of this specification,</td>
</tr>
<tr>
<td></td>
<td>out of Dialog SIP URIs are routed by the Routing algorithm defined in RFC 3261</td>
</tr>
</tbody>
</table>
which is implemented internally by the stack provided that
javax.sip.USE_ROUTER_FOR_ALL_URIS is set to false. In this case,
the installed Router object is consulted for routing decisions pertaining to
non-SIP URIs. An application defined Router object must implement the javax.sip.Router interface.
This property is optional.

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>javax.sip.EXTENSION_METHODS</strong></td>
<td>This configuration value informs the underlying implementation of supported extension methods that create new dialog's. This list must not include methods that are natively supported by this specification such as INVITE, SUBSCRIBE and REFER. This configuration flag should only be used for dialog creating extension methods, other extension methods assuming the implementation understands the method. If more than one method is supported in this property each extension should be separated with a colon for example &quot;FOO:BAR&quot;. This property is optional.</td>
</tr>
</tbody>
</table>

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>javax.sip.RETRANSMISSION_FILTER</strong></td>
<td>Deprecated v1.2. Applications can request retransmission alerts from the ServerTransaction#enableRetransmissionAlerts(). The default retransmission behaviour of this specification is dependent on the application core and is defined as follows:</td>
</tr>
</tbody>
</table>

- User Agent Client: Retransmissions of ACK Requests are the responsibility of the application. All other retransmissions are handled by the SipProvider.
- User Agent Server: Retransmissions of 1xx, 2xx Responses are the responsibility of the application. All other retransmissions are handled by the SipProvider.
<li>Stateful Proxy: As stateful proxies have no Invite transactions all retransmissions are handled by the SipProvider.
<li>Stateless Proxy: As stateless proxies are not transactional all retransmissions are the responsibility of the application and will not be handled the SipProvider.
</ul>
This filter can be viewed as a helper function for User Agents that can be set by an application to prevent the application from handling retransmission of ACK Requests, 1xx and 2xx Responses for INVITE transactions, i.e. the SipProvider will handle the retransmissions. This utility is useful for hiding protocol retransmission semantics from higher level programming environments. The acceptable values are ON/OFF. This property is optional, therefore if not supplied the default is OFF.

```
javax.sip.AUTOMATIC_DIALOG_SUPPORT
```

This property specifies the defined values 'ON' and 'OFF'. The default value is 'ON'. The default behavior represents a common mode of stack operation and allows the construction of simple user agents. This property is optional. This is summarized as:

```
<ul>
<li>A dialog gets created on a dialog creating transaction.</li>
<li>The first response having both a From and a To tag creates the transaction.</li>
<li>The first 2xx response to the transaction will drive the dialog to the CONFIRMED state.</li>
</ul>
```

The ability to turn off dialog support is motivated by dialog free servers (such as proxy servers) that do not want to pay the overhead of the dialog layer and user agents that may want to create multiple dialogs for a single INVITE (as a result of forking by proxy servers). The following behavior is defined when the configuration parameter is set to 'OFF'.

```
<ul type="circle">
<li>The application is responsible to create the Dialog if desired.</li>
<li>The application may create a Dialog and associate it with a response (provisional or final) of a dialog creating request.</li>
</ul>
```

Since v1.2.

```
<table>
<thead>
<tr>
<th>Java Class</th>
<th>Description</th>
</tr>
</thead>
</table>
|javax.sip.FORKABLE_EVENTS| Comma separated list of events for which the implementation should expect forked SUBSCRIBE dialogs. Each element of this list must have the syntax packagename.eventname.

This configuration parameter is provided in order to support the following behavior (defined in RFC 3265):

Successful SUBSCRIBE requests will normally receive only one 200-class response; however, due to forking, the subscription may have been accepted by multiple nodes. The subscriber MUST therefore be prepared to receive NOTIFY requests with "From:" tags which differ from the "To:" tag received in the SUBSCRIBE 200-class response.

If multiple NOTIFY messages are received in different dialogs in response to a single SUBSCRIBE message, each dialog represents a different destination to which the SUBSCRIBE request was forked.

Each event package MUST specify whether forked SUBSCRIBE requests are allowed to install multiple subscriptions. If such behavior is not allowed, the first potential dialog-establishing message will create a dialog.

All subsequent NOTIFY messages which correspond to the SUBSCRIBE message (i.e., match "To","From","From" header "tag" parameter, "Call-ID", "CSeq", "Event", and "Event" header "id" parameter) but which do not match the dialog would be rejected with a 481 response. This property is optional.

Since v1.2

<table>
<thead>
<tr>
<th>Java Class</th>
<th>Description</th>
</tr>
</thead>
</table>
|javax.sip.USE_ROUTER_FOR_ALL_URIS| If set to true then the application installed
* Router is consulted for ALL routing decisions (ie. both out of dialog SIP and non-SIP request URI's -- identical to the behavior supported in v1.1 of this specification). If set to *false* the user installed router will only be consulted for routing of Non-SIP URIs.
* Implementations may thus provide support for sophisticated operations such as DNS lookup for SIP URIs using the procedures defined in RFC 3263 (support for RFC 3263 is not mandatory for this specification). This property is optional.
* The default value for this parameter is *true*.

*<p>*
*<b>Since v1.2.</b>*
*<td>*
*<tr>*
*<table> <center>*
*<@see SipFactory><br/>*<@see SipProvider><br/>*
*<@author BEA Systems, NIST><br/>*<@version 1.2><br/>*
*<*/

Found in path(s):
* /opt/cola/permits/1256060690_1646678993.07/0/jain-sip-api-1-2-sources-2-jar/sip/SipStack.java

No license file was found, but licenses were detected in source scan.

/**
* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
* Unpublished - rights reserved under the Copyright Laws of the United States.
* Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
* Copyright © 2005 BEA Systems, Inc. All rights reserved.
* Use is subject to license terms.
* This distribution may include materials developed by third parties.
* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
*
* Module Name : JSIP Specification
* File Name : AddressFactory.java
* Author : Phelim O'Doherty
* HISTORY
* Version Date Author Comments
* 1.1 08/10/2002 Phelim O'Doherty Initial version
* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
*/
Found in path(s):
* /opt/cola/permits/1256060690_1646678993.07/0/jain-sip-api-1-2-sources-2-jar/sip/address/AddressFactory.java
No license file was found, but licenses were detected in source scan.

/**
 * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 * Unpublished - rights reserved under the Copyright Laws of the United States.
 * Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
 * Copyright © 2005 BEA Systems, Inc. All rights reserved.
 *
 * Use is subject to license terms.
 *
 * This distribution may include materials developed by third parties.
 *
 * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 *
 * Module Name   : JSIP Specification
 * File Name     : PriorityHeader.java
 * Author        : Phelim O'Doherty
 *
 * HISTORY
 * Version Date Author Comments
 * 1.1 08/10/2002 Phelim O'Doherty
 *~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 */

Found in path(s):
* /opt/cola/permits/1256060690_1646678993.07/0/jain-sip-api-1-2-sources-2-jar/sip/header/PriorityHeader.java
No license file was found, but licenses were detected in source scan.

/**
 * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 * Unpublished - rights reserved under the Copyright Laws of the United States.
 * Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
 * Copyright © 2005 BEA Systems, Inc. All rights reserved.
 *
 * Use is subject to license terms.
 *
 * This distribution may include materials developed by third parties.
 *
 * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 *
 * Module Name   : JSIP Specification
 * File Name     : AcceptHeader.java
 * Author        : Phelim O'Doherty
 *
 * HISTORY
 * Version Date Author Comments

Found in path(s):
 */

**
* Unpublished - rights reserved under the Copyright Laws of the United States.
* Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
* Copyright © 2005 BEA Systems, Inc. All rights reserved.
*
* Use is subject to license terms.
*
* This distribution may include materials developed by third parties.
*
* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
*
* Module Name : JSIP Specification
* File Name   : ContentTypeHeader.java
* Author      : Phelim O'Doherty
*
* HISTORY
* Version Date Author   Comments
* 1.1  08/10/2002 Phelim O'Doherty
*
**/

Found in path(s):
 */

**
* Unpublished - rights reserved under the Copyright Laws of the United States.
* Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
* Copyright © 2005 BEA Systems, Inc. All rights reserved.
*
* Use is subject to license terms.
*
* This distribution may include materials developed by third parties.
*
* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
*
* Module Name : JSIP Specification

Open Source Used In security_automation identity-guest-cs 2556
* File Name     : ClientTransaction.java
* Author        : Phelim O'Doherty
*
* HISTORY
* Version   Date      Author              Comments
* 1.1     08/10/2002 Phelim O'Doherty    Initial version
* 1.2     16/06/2005 Phelim O'Doherty    Deprecated createAck method.
*~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 */

Found in path(s):
* /opt/cola/permits/1256060690_1646678993.07/0/jain-sip-api-1-2-sources-2-jar/sip/ClientTransaction.java
No license file was found, but licenses were detected in source scan.

/**
  * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
  * Unpublished - rights reserved under the Copyright Laws of the United States.
  * Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
  * Copyright © 2005 BEA Systems, Inc. All rights reserved.
  *
  * Use is subject to license terms.
  *
  * This distribution may include materials developed by third parties.
  *
  * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
  *
  * Module Name   : JSIP Specification
  * File Name     : SIPIfMatchHeader.java
  * Author        : Jeroen van Bemmel
*
* HISTORY
* Version   Date      Author              Comments
* 1.1     27/10/2005 Jeroen van Bemmel   Initial version, header to support RFC3903.
*~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 */

Found in path(s):
* /opt/cola/permits/1256060690_1646678993.07/0/jain-sip-api-1-2-sources-2-jar/sip/header/SIPIfMatchHeader.java
No license file was found, but licenses were detected in source scan.

/**
  * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
  * Unpublished - rights reserved under the Copyright Laws of the United States.
  * Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
  * Copyright © 2005 BEA Systems, Inc. All rights reserved.
  *
  * Use is subject to license terms.
  * 
  * Module Name   : JSIP Specification
  * File Name     : SIPIfMatchHeader.java
  * Author        : Jeroen van Bemmel
*
* HISTORY
* Version   Date      Author              Comments
* 1.1     27/10/2005 Jeroen van Bemmel   Initial version, header to support RFC3903.
*~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 */
*/

Found in path(s):
* /opt/cola/permits/1256060690_1646678993.07/0/jain-sip-api-1-2-sources-2-jar/sip/ResponseEvent.java
No license file was found, but licenses were detected in source scan.

/****
* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
* Unpublished - rights reserved under the Copyright Laws of the United States.
* Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
* Copyright © 2005 BEA Systems, Inc. All rights reserved.
* Use is subject to license terms.
* This distribution may include materials developed by third parties.
* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
* Module Name : JSIP Specification
* File Name   : TransactionAlreadyExistsException.java
* Author      : Phelim O'Doherty
* HISTORY
* Version   Date      Author              Comments
* 1.1     20/12/2002  Phelim O'Doherty    Initial version
* 1.2     12/15/2004  M. RanganathanAdded getDialog method
* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
*/

Found in path(s):
* 1.1 08/10/2002 Phelim O'Doherty
* 1.2 13/06/2005 Phelim O'Doherty  Deprecated get/set URI parameter
* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
*/

Found in path(s):
* /opt/cola/permits/1256060690_1646678993.07/0/jain-sip-api-1-2-sources-2-jar/sip/header/WWWAuthenticateHeader.java
No license file was found, but licenses were detected in source scan.

/**
 * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 * Unpublished - rights reserved under the Copyright Laws of the United States.
 * Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
 * Copyright © 2005 BEA Systems, Inc. All rights reserved.
 * *
 * Use is subject to license terms.
 *
 * This distribution may include materials developed by third parties.
 *
 * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 * Module Name : JSIP Specification
 * File Name : AlertInfoHeader.java
 * Author : Phelim O'Doherty
 * *
 * HISTORY
 * Version Date Author Comments
 * 1.1 08/10/2002 Phelim O'Doherty
* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
*/

Found in path(s):
* /opt/cola/permits/1256060690_1646678993.07/0/jain-sip-api-1-2-sources-2-jar/sip/header/AlertInfoHeader.java
No license file was found, but licenses were detected in source scan.

/**
 * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 * Unpublished - rights reserved under the Copyright Laws of the United States.
 * Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
 * Copyright © 2005 BEA Systems, Inc. All rights reserved.
 * *
 * Use is subject to license terms.
 *
 * This distribution may include materials developed by third parties.
 *
 * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 *
* Module Name : JSIP Specification
* File Name : MimeVersionHeader.java
* Author : Phelim O'Doherty
*
* HISTORY
* Version     Date      Author              Comments
* 1.1 08/10/2002 Phelim O'Doherty Initial version

* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
*/

Found in path(s):
* /opt/cola/permits/1256060690_1646678993.07/0/jain-sip-api-1-2-sources-2-jar/sip/header/MimeVersionHeader.java

No license file was found, but licenses were detected in source scan.

/**

* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
* Unpublished - rights reserved under the Copyright Laws of the United States.
* Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
* Copyright © 2005 BEA Systems, Inc. All rights reserved.
* 
* Use is subject to license terms.
* 
* This distribution may include materials developed by third parties.
* 
* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
* 
* Module Name : JSIP Specification
* File Name : ContentEncodingHeader.java
* Author : Phelim O'Doherty
*
* HISTORY
* Version     Date      Author              Comments
* 1.1 08/10/2002 Phelim O'Doherty

* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
*/

Found in path(s):
* /opt/cola/permits/1256060690_1646678993.07/0/jain-sip-api-1-2-sources-2-jar/sip/header/ContentEncodingHeader.java

No license file was found, but licenses were detected in source scan.

/**

* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
* Unpublished - rights reserved under the Copyright Laws of the United States.
* Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
* Copyright © 2005 BEA Systems, Inc. All rights reserved.
* 
*
Found in path(s):
* /opt/cola/permits/1256060690_1646678993.07/0/jain-sip-api-1-2-sources-2-jar/sip/header/CallInfoHeader.java

No license file was found, but licenses were detected in source scan.

/**
 * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 * Unpublished - rights reserved under the Copyright Laws of the United States.
 * Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
 * Copyright © 2005 BEA Systems, Inc. All rights reserved.
 * Use is subject to license terms.
 *
 * This distribution may include materials developed by third parties.
 *
 * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 *
 * Module Name : JSIP Specification
 * File Name : SubscriptionStateHeader.java
 * Author : Phelim O'Doherty
 *
 * HISTORY
 * Version Date Author Comments
 * 1.1 13/12/2002 Phelim O'Doherty Initial version, extension header to support RFC3265
 *
 */

Found in path(s):
* /opt/cola/permits/1256060690_1646678993.07/0/jain-sip-api-1-2-sources-2-jar/sip/header/SubscriptıonStateHeader.java

No license file was found, but licenses were detected in source scan.

/**
 * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 * Unpublished - rights reserved under the Copyright Laws of the United States.
 * Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
 * Copyright © 2005 BEA Systems, Inc. All rights reserved.
 * Use is subject to license terms.
 *
 * This distribution may include materials developed by third parties.
 *
 * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 *
 * Module Name : JSIP Specification
 * File Name : TooManyHopsException.java
 * Author : Phelim O'Doherty
 *
Open Source Used In security_automation identity-guest-cs 2566
Found in path(s):
* /opt/cola/permits/1256060690_1646678993.07/0/jain-sip-api-1-2-sources-2-jar/sip/header/RAckHeader.java
No license file was found, but licenses were detected in source scan.

/**
 * Unpublished - rights reserved under the Copyright Laws of the United States.
 * Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
 * Copyright © 2005 BEA Systems, Inc. All rights reserved.
 */

Found in path(s):
* /opt/cola/permits/1256060690_1646678993.07/0/jain-sip-api-1-2-sources-2-jar/sip/header/ExpiresHeader.java
No license file was found, but licenses were detected in source scan.

/**
 * Unpublished - rights reserved under the Copyright Laws of the United States.
 * Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
 * Copyright © 2005 BEA Systems, Inc. All rights reserved.
 */
* Use is subject to license terms.
*
* This distribution may include materials developed by third parties.
*
* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
*
* Module Name   : JSIP Specification
* File Name     : SubjectHeader.java
* Author        : Phelim O'Doherty
*
* HISTORY
* Version  Date  Author    Comments
* 1.1     08/10/2002  Phelim O'Doherty
*~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
*/

Found in path(s):
* /opt/cola/permits/1256060690_1646678993.07/0/jain-sip-api-1-2-sources-2-jar/sip/header/SubjectHeader.java
No license file was found, but licenses were detected in source scan.

/**
 * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 * Unpublished - rights reserved under the Copyright Laws of the United States.
 * Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
 * Copyright © 2005 BEA Systems, Inc. All rights reserved.
 * *
 * Use is subject to license terms.
 * *
 * This distribution may include materials developed by third parties.
 * *
 * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 * *
 * Module Name   : JSIP Specification
 * File Name     : AllowHeader.java
 * Author        : Phelim O'Doherty
 *
 * HISTORY
 * Version  Date  Author    Comments
 * 1.1     08/10/2002  Phelim O'Doherty
 *~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
*/

Found in path(s):
* /opt/cola/permits/1256060690_1646678993.07/0/jain-sip-api-1-2-sources-2-jar/sip/header/AllowHeader.java
No license file was found, but licenses were detected in source scan.

/**
 * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Unpublished - rights reserved under the Copyright Laws of the United States.
Copyright © 2005 BEA Systems, Inc. All rights reserved.

Use is subject to license terms.

This distribution may include materials developed by third parties.

Module Name : JSIP Specification
File Name : TransportAlreadySupportedException.java
Author : Phelim O'Doherty

HISTORY
Version Date Author Comments
1.2 15/08/2005 Phelim O'Doherty Initial version

jar/sip/InvalidArgumentException.java
No license file was found, but licenses were detected in source scan.

/**
 *  Unpublished - rights reserved under the Copyright Laws of the United States.
 *  Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
 *  Copyright © 2005 BEA Systems, Inc. All rights reserved.
 *  
 *  Use is subject to license terms.
 *  
 *  This distribution may include materials developed by third parties.
 *  
 *  ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 *  
 *  Module Name   : JSIP Specification
 *  File Name     : Router.java
 *  Author        : Phelim O'Doherty
 *  
 *  HISTORY
 *  Version   Date      Author              Comments
 *  1.1     19/12/2002  Phelim O'Doherty    Initial version
 *  1.2     16/06/2005  Phelim O'Doherty    Deprecated getNextHops and replaced
 *                        with getNextHop.
 *  M. Ranganathan Worked out details and clarified behavior.
 *  Sarit Galanos
 *  Jeroen van Bemmel
 *  ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 */

Found in path(s):
* /opt/cola/permits/1256060690_1646678993.07/0/jain-sip-api-1-2-sources-2-jar/sip/address/Router.java
No license file was found, but licenses were detected in source scan.

/**
 *  Unpublished - rights reserved under the Copyright Laws of the United States.
 *  Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
 *  Copyright © 2005 BEA Systems, Inc. All rights reserved.
 *  
 *  Use is subject to license terms.
 *  
 *  This distribution may include materials developed by third parties.
 *  
 *  ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 *  
 *  Module Name   : JSIP Specification
 *  File Name     : Encoding.java
* Author        : Phelim O'Doherty

* HISTORY
* Version  Date  Author            Comments
* 1.1     08/10/2002  Phelim O'Doherty

*~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
*/

Found in path(s):
* /opt/cola/permits/1256060690_1646678993.07/0/jain-sip-api-1-2-sources-2-jar/sip/header/Encoding.java
No license file was found, but licenses were detected in source scan.

/**
 * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 * Unpublished - rights reserved under the Copyright Laws of the United States.
 * Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
 * Copyright © 2005 BEA Systems, Inc. All rights reserved.
 * 
 * Use is subject to license terms.
 * 
 * This distribution may include materials developed by third parties.
 * 
 * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 * 
 * Module Name   : JSIP Specification
 * File Name     : RequestEvent.java
 * Author        : Phelim O'Doherty

* HISTORY
* Version  Date  Author            Comments
* 1.1     08/10/2002  Phelim O'Doherty  Initial version
* 1.2 02/15/2005M. RanganathangetDialog get the current dialog.
*~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
*/

Found in path(s):
* /opt/cola/permits/1256060690_1646678993.07/0/jain-sip-api-1-2-sources-2-jar/sip/RequestEvent.java
No license file was found, but licenses were detected in source scan.

/**
 * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 * Unpublished - rights reserved under the Copyright Laws of the United States.
 * Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
 * Copyright © 2005 BEA Systems, Inc. All rights reserved.
 * 
 * Use is subject to license terms.
 * 
 * Module Name   : JSIP Specification
 * File Name     : RequestEvent.java
 * Author        : Phelim O'Doherty

* HISTORY
* Version  Date  Author            Comments
* 1.1     08/10/2002  Phelim O'Doherty  Initial version
* 1.2 02/15/2005M. RanganathangetDialog get the current dialog.
*~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
*/
* This distribution may include materials developed by third parties.
* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
* Module Name   : JSIP Specification
* File Name     : Transaction.java
* Author        : Phelim O'Doherty
* HISTORY
* Version   Date       Author                  Comments
* 1.1     08/10/2002 Phelim O'Doherty Initial version
* 1.2     12/15/2004 M. Ranganathan Clarified behavior of getDialog when
* Phelim O'Doherty AUTOMATIC_DIALOG_SUPPORT is set to off.
* Added two methods - set/getApplicationData
* Added terminate method.
*~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
*/

Found in path(s):
* /opt/cola/permits/1256060690_1646678993.07/0/jain-sip-api-1-2-sources-2-jar/sip/Transaction.java
No license file was found, but licenses were detected in source scan.

/**
* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
* Unpublished - rights reserved under the Copyright Laws of the United States.
* Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
* Copyright © 2005 BEA Systems, Inc. All rights reserved.
* Use is subject to license terms.
* This distribution may include materials developed by third parties.
* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
* Module Name   : JSIP Specification
* File Name     : MediaType.java
* Author        : Phelim O'Doherty
* HISTORY
* Version   Date       Author                  Comments
* 1.1     08/10/2002 Phelim O'Doherty
*~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
*/

Found in path(s):
* /opt/cola/permits/1256060690_1646678993.07/0/jain-sip-api-1-2-sources-2-jar/sip/header/MediaType.java
No license file was found, but licenses were detected in source scan.
/**
 * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 * Unpublished - rights reserved under the Copyright Laws of the United States.
 * Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
 * Copyright © 2005 BEA Systems, Inc. All rights reserved.
 * Use is subject to license terms.
 * This distribution may include materials developed by third parties.
 * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 *
 * Module Name   : JSIP Specification
 * File Name     : ProxyAuthorizationHeader.java
 * Author        : Phelim O'Doherty
 *
 * HISTORY
 * Version   Date      Author              Comments
 * 1.1     08/10/2002  Phelim O'Doherty
 * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 */

Found in path(s):
* /opt/cola/permits/1256060690_1646678993.07/jain-sip-api-1-2-sources-2.jar/sip/header/ProxyAuthorizationHeader.java

No license file was found, but licenses were detected in source scan.

/**
 * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 * Unpublished - rights reserved under the Copyright Laws of the United States.
 * Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
 * Copyright © 2005 BEA Systems, Inc. All rights reserved.
 * Use is subject to license terms.
 * This distribution may include materials developed by third parties.
 * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 *
 * Module Name   : JSIP Specification
 * File Name     : EventHeader.java
 * Author        : Phelim O'Doherty
 *
 * HISTORY
 * Version   Date      Author              Comments
 * 1.1     13/12/2002  Phelim O'Doherty    Initial version, extension header to support RFC3265
 * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Found in path(s):
* /opt/cola/permits/1256060690_1646678993.07/0/jain-sip-api-1-2-sources-2-jar/sip/header/EventHeader.java
No license file was found, but licenses were detected in source scan.

/**
 * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 * Unpublished - rights reserved under the Copyright Laws of the United States.
 * Copyright © 2003 Sun Microsystems, Inc. All rights reserved.
 * Copyright © 2005 BEA Systems, Inc. All rights reserved.
 * *
 * Use is subject to license terms.
 * *
 * This distribution may include materials developed by third parties.
 * *
 * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 * *
 * Module Name  : JSIP Specification
 * File Name    : ObjectInUseException.java
 * Author       : Phelim O'Doherty
 * *
 * HISTORY
 * Version   Date      Author              Comments
 * 1.1 08/10/2002 Phelim O'Doherty  Initial version
 * ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 */

Found in path(s):
* /opt/cola/permits/1256060690_1646678993.07/0/jain-sip-api-1-2-sources-2-jar/ObjectInUseException.java

1.252 netty-handler 4.1.69.Final
1.252.1 Available under license :
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014 The Netty Project
 * *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * *
 * https://www.apache.org/licenses/LICENSE-2.0
 * *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT

Open Source Used In security_automation identity-guest-cs 2574
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
jar/io/netty/handler/traffic/GlobalChannelTrafficCounter.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
jar/io/netty/handler/ssl/util/LazyX509Certificate.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
jar/io/netty/handler/ssl/JdkBaseApplicationProtocolNegotiator.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
jar/io/netty/handler/ipfilter/IpFilterRule.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
jar/io/netty/handler/ssl/IdentityCipherSuiteFilter.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
jar/io/netty/handler/ssl/Java7SslParametersUtils.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
jar/io/netty/handler/ssl/OpenSslServerSessionContext.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
jar/io/netty/handler/ssl/JdkSslEngine.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
jar/io/netty/handler/ssl/OpenJdkSelfSignedCertGenerator.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
jar/io/netty/handler/ssl/JdkApplicationProtocolNegotiator.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
jar/io/netty/handler/ssl/SslContext.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
jar/io/netty/handler/ssl/OpenSslApplicationProtocolNegotiator.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
jar/io/netty/handler/ssl/ApplicationProtocolConfig.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
jar/io/netty/handler/ssl/OpenSslSessionContext.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
jar/io/netty/handler/ssl/JdkSslClientContext.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
jar/io/netty/handler/ssl/util/ThreadLocalInsecureRandom.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
jar/io/netty/handler/ipfilter/AbstractRemoteAddressFilter.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
jar/io/netty/handler/ssl/CipherSuiteFilter.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
jar/io/netty/handler/ssl/JettyAlpnSslEngine.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
jar/io/netty/handler/ipfilter/IpSubnetFilterRule.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
jar/io/netty/handler/ssl/OpenSslContext.java

Open Source Used In security_automation identity-guest-cs 2575
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-jar/io/netty/handler/ssl/SslUtils.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-jar/io/netty/handler/ssl/util/BouncyCastleSelfSignedCertGenerator.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-jar/io/netty/handler/ssl/SniHandler.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-jar/io/netty/handler/ipfilter/IpFilterRuleType.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-jar/io/netty/handler/ssl/IpFilter/IPFilterRuleType.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-jar/io/netty/handler/ssl/IpFilter/IPFilterRuleType.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-jar/io/netty/handler/ssl/InsecureTrustManagerFactory.java
No license file was found, but licenses were detected in source scan.

~ Copyright 2012 The Netty Project
~
~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~
~ https://www.apache.org/licenses/LICENSE-2.0
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.

Found in path(s):

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2019 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * 
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
Found in path(s):
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-jar/io/netty/handler/address/DynamicAddressConnectHandler.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-jar/io/netty/handler/ssl/util/TrustManagerFactoryWrapper.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-jar/io/netty/handler/ssl/util/X509KeyManagerWrapper.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-jar/io/netty/handler/ssl/SslMasterKeyHandler.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-jar/io/netty/handler/address/package-info.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-jar/io/netty/handler/ssl/util/KeyManagerFactoryWrapper.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-jar/io/netty/handler/ssl/util/SimpleKeyManagerFactory.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2011 The Netty Project
 * 
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * 
 * https://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2018 The Netty Project
 * 
 * The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* 
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
  jar/io/netty/handler/ssl/ExtendedOpenSslSession.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
  jar/io/netty/handler/ssl/OpenSslX509TrustManagerWrapper.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
  jar/io/netty/handler/ssl/OpenSslKeyMaterial.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
  jar/io/netty/handler/ssl/OpenSslSession.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
  jar/io/netty/handler/ssl/SignatureAlgorithmConverter.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
  jar/io/netty/handler/ssl/OpenSslCachingKeyMaterialProvider.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
  jar/io/netty/handler/ssl/OpenSslCachingX509KeyManagerFactory.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
  jar/io/netty/handler/ssl/DefaultOpenSslKeyMaterial.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
  jar/io/netty/handler/ssl/OpenSslX509KeyManagerFactory.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
  jar/io/netty/handler/ssl/OpenSslKeyMaterialProvider.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
  jar/io/netty/handler/ssl/OpenSslPrivateKey.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2013 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* 
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
  jar/io/netty/handler/ssl/SslHandshakeCompletionEvent.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2016 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License, version
 * 2.0 (the "License"); you may not use this file except in compliance with the
 * License. You may obtain a copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
  jar/io/netty/handler/flow/FlowControlHandler.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2012 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):

/*
 * Copyright 2014 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

// Try the OpenJDK's proprietary implementation.
/*
 * Copyright 2021 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-jar/io/netty/handler/ssl/AsyncRunnable.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-jar/io/netty/handler/ssl/Ciphers.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-jar/io/netty/handler/ssl/GroupsConverter.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-jar/io/netty/handler/ssl/OpenSslSessionId.java
"
No license file was found, but licenses were detected in source scan.

# The Netty Project licenses this file to you under the Apache License,
# version 2.0 (the "License"); you may not use this file except in compliance
# with the License. You may obtain a copy of the License at:
# distributed under the License is distributed on an "AS IS" BASIS, WITHOUT

Found in path(s):
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2016 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-jar/io/netty/handler/ssl/util/X509TrustManagerWrapper.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-jar/io/netty/handler/ssl/PemEncoded.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-jar/io/netty/handler/ssl/PemX509Certificate.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-jar/io/netty/handler/ssl/OpenSslKeyMaterialManager.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-jar/io/netty/handler/flush/FlushConsolidationHandler.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
jar/io/netty/handler/ssl/PemPrivateKey.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-jar/io/netty/handler/ssl/PemValue.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-jar/io/netty/handler/ssl/Java8SslUtils.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-jar/io/netty/handler/ssl/ClientAuth.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-jar/io/netty/handler/ssl/OpenSslSessionTicketKey.java

No license file was found, but licenses were detected in source scan.
/*
 * Copyright 2017 The Netty Project
 * 
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * 
 * https://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
  /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
jar/io/netty/handler/ssl/ocsp/OcspClientHandler.java
  /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
jar/io/netty/handler/ssl/AbstractSniHandler.java
  /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
jar/io/netty/handler/ssl/SniCompletionEvent.java
  /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
jar/io/netty/handler/ssl/ocsp/package-info.java
  /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
jar/io/netty/handler/ssl/Conscrypt.java
  /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
jar/io/netty/handler/ssl/SslClientHelloHandler.java
  /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
jar/io/netty/handler/ssl/OptionalSslHandler.java
  /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
jar/io/netty/handler/ssl/SslCloseCompletionEvent.java
  /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
jar/io/netty/handler/ssl/JdkAlpnSslUtils.java
  /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
jar/io/netty/handler/ssl/ConscryptAlpnSslEngine.java
  /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
jar/io/netty/handler/ssl/SslCompletionEvent.java
  /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
jar/io/netty/handler/ssl/JdkAlpnSslEngine.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2020 The Netty Project
 * 
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance

* with the License. You may obtain a copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
  jar/io/netty/handler/pcap/PcapHeaders.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
  jar/io/netty/handler/pcap/package-info.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
  jar/io/netty/handler/pcap/PcapWriteHandler.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
  jar/io/netty/handler/pcap/PcapWriter.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
  jar/io/netty/handler/pcap/UDPPacket.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
  jar/io/netty/handler/pcap/EthernetPacket.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
  jar/io/netty/handler/pcap/TCPPacket.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
  jar/io/netty/handler/pcap/IPPacket.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2012 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
jar/io/netty/handler/logging/LoggingHandler.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
jar/io/netty/handler/stream/ChunkedNioFile.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
jar/io/netty/handler/logging/LogLevel.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
jar/io/netty/handler/stream/ChunkedFile.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
jar/io/netty/handler/timeout/ReadTimeoutException.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
jar/io/netty/handler/ssl/NotSslRecordException.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
jar/io/netty/handler/timeout/IdleStateHandler.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
jar/io/netty/handler/ssl/SslHandler.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
jar/io/netty/handler/timeout/IdleState.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
jar/io/netty/handler/stream/ChunkedInput.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
jar/io/netty/handler/ssl/util/package-info.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
jar/io/netty/handler/timeout/WriteTimeoutException.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
jar/io/netty/handler/logging/package-info.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
jar/io/netty/handler/timeout/ReadTimeoutHandler.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
jar/io/netty/handler/stream/ChunkedWriteHandler.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
jar/io/netty/handler/timeout/WriteTimeoutHandler.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
jar/io/netty/handler/stream/package-info.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
jar/io/netty/handler/timeout/package-info.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
jar/io/netty/handler/timeout/IdleStateEvent.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
jar/io/netty/handler/stream/ChunkedStream.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
jar/io/netty/handler/ssl/package-info.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
jar/io/netty/handler/stream/ChunkedNioStream.java
* /opt/cola/permits/1265673641_1643932611.53/0/netty-handler-4-1-69-final-sources-
jar/io/netty/handler/timeout/TimeoutException.java

No license file was found, but licenses were detected in source scan.

/*
1.253 asm-util 7.1

1.253.1 Available under license:

No license file was found, but licenses were detected in source scan.

// All rights reserved.
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions
// are met:
// 1. Redistributions of source code must retain the above copyright
//    notice, this list of conditions and the following disclaimer.
// 2. Redistributions in binary form must reproduce the above copyright
//    notice, this list of conditions and the following disclaimer in the
//    documentation and/or other materials provided with the distribution.
// 3. Neither the name of the copyright holders nor the names of its
//    this software without specific prior written permission.
Found in path(s):
* /opt/cola/permits/1135892376_1613626859.73/0/asm-util-7-1-sources-2-jar/org/objectweb/asm/util/CheckFieldAdapter.java
* /opt/cola/permits/1135892376_1613626859.73/0/asm-util-7-1-sources-2-jar/org/objectweb/asm/util/TraceMethodVisitor.java
* /opt/cola/permits/1135892376_1613626859.73/0/asm-util-7-1-sources-2-jar/org/objectweb/asm/util/CheckMethodAdapter.java
* /opt/cola/permits/1135892376_1613626859.73/0/asm-util-7-1-sources-2-jar/org/objectweb/asm/util/CheckSignatureAdapter.java
* /opt/cola/permits/1135892376_1613626859.73/0/asm-util-7-1-sources-2-jar/org/objectweb/asm/util/TraceFieldVisitor.java
* /opt/cola/permits/1135892376_1613626859.73/0/asm-util-7-1-sources-2-jar/org/objectweb/asm/util/Printer.java
* /opt/cola/permits/1135892376_1613626859.73/0/asm-util-7-1-sources-2-jar/org/objectweb/asm/util/CheckModuleAdapter.java
* /opt/cola/permits/1135892376_1613626859.73/0/asm-util-7-1-sources-2-jar/org/objectweb/asm/util/CheckAnnotationAdapter.java
* /opt/cola/permits/1135892376_1613626859.73/0/asm-util-7-1-sources-2-jar/org/objectweb/asm/util/TraceAnnotationVisitor.java
* /opt/cola/permits/1135892376_1613626859.73/0/asm-util-7-1-sources-2-jar/org/objectweb/asm/util/Textifier.java
* /opt/cola/permits/1135892376_1613626859.73/0/asm-util-7-1-sources-2-jar/org/objectweb/asm/util/TraceSignatureVisitor.java
* /opt/cola/permits/1135892376_1613626859.73/0/asm-util-7-1-sources-2-jar/org/objectweb/asm/util/TraceModuleVisitor.java
* /opt/cola/permits/1135892376_1613626859.73/0/asm-util-7-1-sources-2-jar/org/objectweb/asm/util/ASMifier.java
* /opt/cola/permits/1135892376_1613626859.73/0/asm-util-7-1-sources-2-jar/org/objectweb/asm/util/TraceClassVisitor.java
* /opt/cola/permits/1135892376_1613626859.73/0/asm-util-7-1-sources-2-jar/org/objectweb/asm/util/CheckClassAdapter.java

No license file was found, but licenses were detected in source scan.

2011 INRIA, France Telecom
* All rights reserved.

* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in the
* documentation and/or other materials provided with the distribution.
* 3. Neither the name of the copyright holders nor the names of its
* contributors may be used to endorse or promote products derived from
* this software without specific prior written permission.

* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
* LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
* CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
* SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
* INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
* CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
* THE POSSIBILITY OF SUCH DAMAGE.

Found in path(s):
* /opt/cola/permits/1135892376_1613626859.73/0/asm-util-7-1-sources-2-jar/org/objectweb/asm/util/package.html

No license file was found, but licenses were detected in source scan.

/**
* ASM: a very small and fast Java bytecode manipulation framework Copyright (c) 2000-2011 INRIA,
* France Telecom All rights reserved.
* *
* Redistributions and use in source and binary forms, with or without modification, are permitted
* provided that the following conditions are met: 1. Redistributions of source code must retain the
* above copyright notice, this list of conditions and the following disclaimer. 2. Redistributions
* in binary form must reproduce the above copyright notice, this list of conditions and the
* following disclaimer in the documentation and/or other materials provided with the distribution.
* 3. Neither the name of the copyright holders nor the names of its contributors may be used to
* endorse or promote products derived from this software without specific prior written permission.
* *
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
* AND ANY EXPRESS OR
* IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
* MERCHANTABILITY AND
* FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
* OWNER OR
* CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY,
* OR CONSEQUENTIAL
* DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
* SERVICES; LOSS OF USE,
* DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF
* LIABILITY,
* WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR
* OTHERWISE) ARISING IN ANY
* WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH
* DAMAGE.
*/

Found in path(s):
* /opt/cola/permits/1135892376_1613626859.73/0/asm-util-7-1-sources-2-jar/org/objectweb/asm/util/ASMifierSupport.java
* /opt/cola/permits/1135892376_1613626859.73/0/asm-util-7-1-sources-2-
1.254 logging-interceptor 3.14.9

1.254.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2015 Square, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1199746110_1630577833.08/0/logging-interceptor-3-14-9-sources-1-jar/okhttp3/logging/HttpLoggingInterceptor.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2018 Square, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
1.255 jldap 2013.08.30.1433

1.255.1 Available under license:

Copyright 1998-2005 The OpenLDAP Foundation
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted only as authorized by the OpenLDAP
Public License.

A copy of this license is available in the file LICENSE in the
top-level directory of the distribution or, alternatively, at

OpenLDAP is a registered trademark of the OpenLDAP Foundation.

Individual files and/or contributed packages may be copyright by
other parties and subject to additional restrictions.

This work is derived from the University of Michigan LDAP v3.3
distribution. Information concerning this software is available
at <http://www.umich.edu/~dirsvcs/ldap/>.

This work also contains materials derived from public sources.

Additional information about OpenLDAP can be obtained at
<http://www.openldap.org/>.

---

Portions Copyright (C) 1999 - 2003 Novell, Inc. All Rights Reserved.

THIS WORK IS SUBJECT TO U.S. AND INTERNATIONAL COPYRIGHT LAWS AND
TREATIES. USE, MODIFICATION, AND REDISTRIBUTION OF THIS WORK IS SUBJECT
TO VERSION 2.0.1 OF THE OPENLDAP PUBLIC LICENSE, A COPY OF WHICH IS
AVAILABLE AT HTTP://WWW.OPENLDAP.ORG/LICENSE.HTML OR IN THE FILE
"LICENSE-2.0.1" IN THE TOP-LEVEL DIRECTORY OF THE DISTRIBUTION.
ANY USE OR EXPLOITATION OF THIS WORK OTHER THAN AS AUTHORIZED IN
VERSION 2.0.1 OF THE OPENLDAP PUBLIC LICENSE, OR OTHER PRIOR WRITTEN
CONSENT FROM NOVELL, COULD SUBJECT THE PERPETRATOR TO CRIMINAL AND
CIVIL LIABILITY.

The OpenLDAP Public License
Version 2.8, 17 August 2003
Redistribution and use of this software and associated documentation ("Software"), with or without modification, are permitted provided that the following conditions are met:

1. Redistributions in source form must retain copyright statements and notices,

2. Redistributions in binary form must reproduce applicable copyright statements and notices, this list of conditions, and the following disclaimer in the documentation and/or other materials provided with the distribution, and

3. Redistributions must contain a verbatim copy of this document.

The OpenLDAP Foundation may revise this license from time to time. Each revision is distinguished by a version number. You may use this Software under terms of this license revision or under the terms of any subsequent revision of the license.

THIS SOFTWARE IS PROVIDED BY THE OPENLDAP FOUNDATION AND ITS CONTRIBUTORS "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE OPENLDAP FOUNDATION, ITS CONTRIBUTORS, OR THE AUTHOR(S) OR OWNER(S) OF THE SOFTWARE BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The names of the authors and copyright holders must not be used in advertising or otherwise to promote the sale, use or other dealing in this Software without specific, written prior permission. Title to copyright in this Software shall at all times remain with copyright holders.

OpenLDAP is a registered trademark of the OpenLDAP Foundation.

Copyright 1999-2003 The OpenLDAP Foundation, Redwood City, California, USA. All Rights Reserved. Permission to copy and distribute verbatim copies of this document is granted.
1.256 jul-to-slf4j 1.7.32

1.256.1 Available under license:

No license file was found, but licenses were detected in source scan.

/**
 * Copyright (c) 2004-2011 QOS.ch
 * All rights reserved.
 *
 * Permission is hereby granted, free of charge, to any person obtaining
 * a copy of this software and associated documentation files (the
 * "Software"), to deal in the Software without restriction, including
 * without limitation the rights to use, copy, modify, merge, publish,
 * distribute, sublicense, and/or sell copies of the Software, and to
 * permit persons to whom the Software is furnished to do so, subject to
 * the following conditions:
 *
 * The above copyright notice and this permission notice shall be
 * included in all copies or substantial portions of the Software.
 *
 * THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
 * EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
 * MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
 * NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
 * LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
 * OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
 * WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
 * /

Found in path(s):
* /opt/cola/permits/1182802247_1629961496.53/0/jul-to-slf4j-1-7-32-sources-jar/org/slf4j/bridge/SLF4JBridgeHandler.java

1.257 jnr-x86asm 1.0.2

1.257.1 Available under license:

Copyright (C) 2010 Wayne Meissner
Copyright (c) 2008-2009, Petr Kobalicek <kobalicek.petr@gmail.com>

Permission is hereby granted, free of charge, to any person
obtaining a copy of this software and associated documentation
files (the "Software"), to deal in the Software without
restriction, including without limitation the rights to use,
copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the
Software is furnished to do so, subject to the following
conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES
OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT
HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY,
WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR
OTHER DEALINGS IN THE SOFTWARE.

1.258 spring-security-jwt 1.0.10.RELEASE
1.258.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2006-2011 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance
 * with
 * the License. You may obtain a copy of the License at
 * *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on
 * an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the
 * specific language governing permissions and limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1136823312_1613930815.78/0/spring-security-jwt-1-0-10-release-sources-1-jar/org/springframework/security/jwt/ JwtHelper.java
* /opt/cola/permits/1136823312_1613930815.78/0/spring-security-jwt-1-0-10-release-sources-1-jar/org/springframework/security/jwt/crypto/sign/Signer.java
* /opt/cola/permits/1136823312_1613930815.78/0/spring-security-jwt-1-0-10-release-sources-1-jar/org/springframework/security/jwt/crypto/sign/RsaKeyHelper.java
* /opt/cola/permits/1136823312_1613930815.78/0/spring-security-jwt-1-0-10-release-sources-1-
** Base64 encoder which is a reduced version of Robert Harder's public domain implementation (version 2.3.7).
* See <a href="http://iharder.net/base64">http://iharder.net/base64</a> for more information.
* 
* @author Luke Taylor
*/

Found in path(s):
* /opt/cola/permits/1136823312_1613930815.78/0/spring-security-jwt-1-0-10-release-sources-1-jar/org/springframework/security/jwt/codec/Base64Codec.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2002-2017 the original author or authors.
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
*   http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
1.259 bouncy-castle 1.68
1.259.1 Available under license:

Copyright (c) 2000-2021 The Legion of the Bouncy Castle Inc. (https://www.bouncycastle.org)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.0

1. Definitions.

1.1. Contributor. means each individual or entity that creates or contributes to the creation of Modifications.

1.2. Contributor Version. means the combination of the Original Software, prior Modifications used by a
Contributor (if any), and the Modifications made by that particular Contributor.

1.3. Covered Software. means (a) the Original Software, or (b) Modifications, or (c) the combination of files
containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. Executable. means the Covered Software in any form other than Source Code.

1.5. Initial Developer. means the individual or entity that first makes Original Software available under this
License.

1.6. Larger Work. means a work which combines Covered Software or portions thereof with code not governed by
the terms of this License.

1.7. License. means this document.

1.8. Licensable. means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. Modifications. means the Source Code and Executable form of any of the following:

   A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

   B. Any new file that contains any part of the Original Software or previous Modification; or

   C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. Original Software. means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. Patent Claims. means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code. means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. You. (or .Your.) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, .You. includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, .control. means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.
(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial
3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients' rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.


You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.

Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.
5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as Participant) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.
8. U.S. GOVERNMENT END USERS.

The Covered Software is a .commercial item., as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of .commercial computer software. (as that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and .commercial computer software documentation, as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)

The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc. 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not
Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its
scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under
the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.
It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF
THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER
PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to
achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to
most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a
pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.

Copyright (C)

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public
License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later
version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the
implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General
Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the
Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and
you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License.
Of course, the commands you use may be called something other than `show w' and `show c'; they could even be
mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright
disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at
compilers) written by James Hacker.
This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

"CLASSPATH" EXCEPTION TO THE GPL VERSION 2

Certain source files distributed by Sun Microsystems, Inc. are subject to the following clarification and special exception to the GPL Version 2, but only where Sun has expressly included in the particular source file's header the words

"Sun designates this particular file as subject to the "Classpath" exception as provided by Sun in the License file that accompanied this code."

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License Version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

JUnit

Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and

b) in the case of each subsequent Contributor:

i) changes to the Program, and

ii) additions to the Program;

where such changes and/or additions to the Program originate from and are
distributed by that particular Contributor. A Contribution 'originates' from a
Contributor if it was added to the Program by such Contributor itself or anyone
acting on such Contributor's behalf. Contributions do not include additions to
the Program which: (i) are separate modules of software distributed in
conjunction with the Program under their own license agreement, and (ii) are
not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents " mean patent claims licensable by a Contributor which are
necessarily infringed by the use or sale of its Contribution alone or when
combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement,
including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants
Recipient a non-exclusive, worldwide, royalty-free copyright license to
reproduce, prepare derivative works of, publicly display, publicly perform,
distribute and sublicense the Contribution of such Contributor, if any, and
such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants
Recipient a non-exclusive, worldwide, royalty-free patent license under
Licensed Patents to make, use, sell, offer to sell, import and otherwise
transfer the Contribution of such Contributor, if any, in source code and
object code form. This patent license shall apply to the combination of the
Contribution and the Program if, at the time the Contribution is added by the
Contributor, such addition of the Contribution causes such combination to be
covered by the Licensed Patents. The patent license shall not apply to any
other combinations which include the Contribution. No hardware per se is
licensed hereunder.

c) Recipient understands that although each Contributor grants the
licenses to its Contributions set forth herein, no assurances are provided by
any Contributor that the Program does not infringe the patent or other
intellectual property rights of any other entity. Each Contributor disclaims
any liability to Recipient for claims brought by any other entity based on
infringement of intellectual property rights or otherwise. As a condition to
exercising the rights and licenses granted hereunder, each Recipient hereby
assumes sole responsibility to secure any other intellectual property rights
needed, if any. For example, if a third party patent license is required to
allow Recipient to distribute the Program, it is Recipient's responsibility to
acquire that license before distributing the Program.
d) Each Contributor represents that to its knowledge it has sufficient
copyright rights in its Contribution, if any, to grant the copyright license
set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under
its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and

b) its license agreement:

i) effectively disclaims on behalf of all Contributors all warranties and
conditions, express and implied, including warranties or conditions of title
and non-infringement, and implied warranties or conditions of merchantability
and fitness for a particular purpose;

ii) effectively excludes on behalf of all Contributors all liability for
damages, including direct, indirect, special, incidental and consequential
damages, such as lost profits;

iii) states that any provisions which differ from this Agreement are
offered by that Contributor alone and not by any other party; and

iv) states that source code for the Program is available from such
Contributor, and informs licensees how to obtain it in a reasonable manner on
or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and

b) a copy of this Agreement must be included with each copy of the
Program.

Contributors may not remove or alter any copyright notices contained within the
Program.

Each Contributor must identify itself as the originator of its Contribution, if
any, in a manner that reasonably allows subsequent Recipients to identify the
originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with
respect to end users, business partners and the like. While this license is
intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS
7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.
1.260 nimbus-jose-jwt 8.20
1.260.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*! 
 * nimbus-jose-jwt 
 * 
 * Copyright 2012-2020, Connect2id Ltd. 
 * 
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use 
 * this file except in compliance with the License. You may obtain a copy of the 
 * License at 
 * 
 * http://www.apache.org/licenses/LICENSE-2.0 
 * 
 * Unless required by applicable law or agreed to in writing, software distributed 
 * under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR 
 * CONDITIONS OF ANY KIND, either express or implied. See the License for the 
 * specific language governing permissions and limitations under the License. 
 */

Found in path(s):
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1-
  jar/com/nimbusds/jose/crypto/factories/DefaultJWSSignerFactory.java

No license file was found, but licenses were detected in source scan.

/*! 
 * nimbus-jose-jwt 
 * 
 * Copyright 2012-2018, Connect2id Ltd. 
 * 
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use 
 * this file except in compliance with the License. You may obtain a copy of the 
 * License at 
 * 
 * http://www.apache.org/licenses/LICENSE-2.0 
 * 
 * Unless required by applicable law or agreed to in writing, software distributed 
 * under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR 
 * CONDITIONS OF ANY KIND, either express or implied. See the License for the 
 * specific language governing permissions and limitations under the License. 
 */

Found in path(s):
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1-
jar/com/nimbusds/jose/crypto/Ed25519Verifier.java
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1-
jar/com/nimbusds/jose/crypto/Ed25519Signer.java
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1-
jar/com/nimbusds/jose/util/Base64Codec.java
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1-
jar/com/nimbusds/jose/jwk/JWKSet.java
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1-
jar/com/nimbusds/jose/crypto/X25519Decrypter.java
No license file was found, but licenses were detected in source scan.

/*
* nimbus-jose-jwt
* *
* Copyright 2012-2019, Connect2id Ltd and contributors.
* *
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use
* this file except in compliance with the License. You may obtain a copy of the
* License at
* *
* http://www.apache.org/licenses/LICENSE-2.0
* *
* Unless required by applicable law or agreed to in writing, software distributed
* under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR
* CONDITIONS OF ANY KIND, either express or implied. See the License for the
* specific language governing permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1-
jar/com/nimbusds/jose/crypto/impl/ECDHCryptoProvider.java
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1-
jar/com/nimbusds/jose/proc/JOSEObjectTypeVerifier.java
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1-
jar/com/nimbusds/jose/jwk/source/JWKSecurityContextJWKSet.java
No license file was found, but licenses were detected in source scan.

/*
* nimbus-jose-jwt
* *
* Copyright 2012-2016, Connect2id Ltd and contributors.
* *
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use
* this file except in compliance with the License. You may obtain a copy of the
* License at
* *
* http://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software distributed
  * under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR
  * CONDITIONS OF ANY KIND, either express or implied. See the License for the
  * specific language governing permissions and limitations under the License.

*/

Found in path(s):
  * /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1-
    jar/com/nimbusds/jose/cryptom/pl/CompositeKey.java
  * /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1-
    jar/com/nimbusds/jose/cryptom/pl/RSASSA.java
  * /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1-
    jar/com/nimbusds/jose/cryptom/pl/CriticalHeaderParamsDeferral.java
  * /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1-
    jar/com/nimbusds/jose/cryptom/pl/HMAC.java
  * /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1-
    jar/com/nimbusds/jose/cryptom/pl/PBKDF2.java
  * /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1-
    jar/com/nimbusds/jose/jwk/JWKException.java
  * /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1-
    jar/com/nimbusds/jose/uti/ls/BoundedInputstream.java
  * /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1-
    jar/com/nimbusds/jose/uti/ls/Container.java
  * /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1-
    jar/com/nimbusds/jose/cryptom/pl/RSA1_5.java
  * /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1-
    jar/com/nimbusds/jose/cryptom/pl/ECDSAProvider.java
  * /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1-
    jar/com/nimbusds/jose/cryptom/pl/utils/package-info.java
  * /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1-
    jar/com/nimbusds/jose/jwk/PEMEncodedKeyParser.java
  * /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1-
    jar/com/nimbusds/jose/cryptom/pl/AlgorithmParametersHelper.java
  * /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1-
    jar/com/nimbusds/jose/cryptom/pl/DirectCryptoProvider.java
  * /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1-
    jar/com/nimbusds/jose/jwk/source/JWKSetCache.java
  * /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1-
    jar/com/nimbusds/jose/proc/DefaultJOSEObjectTypeVerifier.java
  * /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1-
    jar/com/nimbusds/jose/cryptom/pl/RSASSAProvider.java
  * /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1-
    jar/com/nimbusds/jose/uti/ls/KeyUsils.java
  * /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1-
    jar/com/nimbusds/jose/jwk/Curve.java
  * /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1-
    jar/com/nimbusds/jose/cryptom/pl/AESGCMKW.java
  * /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1-
No license file was found, but licenses were detected in source scan.


Found in path(s):
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1-jar/com/nimbusds/jose/util/IntegerOverflowException.java
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1-jar/com/nimbusds/jose/jwk/gen/RSAKeyGenerator.java
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1-jar/com/nimbusds/jose/crypto/impl/CipherHelper.java
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1-jar/com/nimbusds/jose/crypto/impl/RSACryptoProvider.java
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1-jar/com/nimbusds/jose/crypto/impl/JWSSignerFactory.java
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1-jar/com/nimbusds/jose/crypto/impl/PasswordBasedCryptoProvider.java
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1-jar/com/nimbusds/jose/crypto/impl/DeflateHelper.java
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1-jar/com/nimbusds/jose/crypto/impl/AAD.java
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1-jar/com/nimbusds/jose/crypto/impl/AESGCM.java
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1-jar/com/nimbusds/jose/crypto/impl/AESCBC.java
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1-jar/com/nimbusds/jose/jwk/source/DefaultJWKSetCache.java
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1-jar/com/nimbusds/jose/crypto/impl/AlgorithmSupportMessage.java
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1-jar/com/nimbusds/jose/crypto/impl/LegacyAESGCM.java
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1-jar/com/nimbusds/jwt/proc/JWTClaimsSetAwareJWSKeySelector.java
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1-jar/com/nimbusds/jose/util/IOUtils.java
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1-jar/com/nimbusds/jose/crypto/impl/package-info.java
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1-jar/com/nimbusds/jose/produce/package-info.java
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1-jar/com/nimbusds/jose/crypto/impl/AESKW.java
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1-jar/com/nimbusds/jose/crypto/impl/ECDSA.java
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1-jar/com/nimbusds/jose/jwk/OctetKeyPair.java
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1-jar/com/nimbusds/jose/crypto/impl/LegacyConcatKDF.java

Open Source Used In security_automation identity-guest-cs 2617
No license file was found, but licenses were detected in source scan.

/**
 * nimbus-jose-jwt
 *
 * Copyright 2012-2019, Connect2id Ltd.
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use
 * this file except in compliance with the License. You may obtain a copy of the
 * License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed
 * under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR
 * CONDITIONS OF ANY KIND, either express or implied. See the License for the
 * specific language governing permissions and limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1-jar/com/nimbusds/jose/proc/JWSAlgorithmFamilyJWSKeySelector.java
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1-jar/com/nimbusds/jwt/proc/DefaultJWTProcessor.java
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1-jar/com/nimbusds/jose/proc/DefaultJOSEProcessor.java
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1-jar/com/nimbusds/jose/proc/JWSVerificationKeySelector.java
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1-jar/com/nimbusds/jose/proc/JWEDecryptionKeySelector.java
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1-jar/com/nimbusds/jose/proc/JOSEProcessorConfiguration.java
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1-jar/com/nimbusds/jose/crypto/ECDHEncrypter.java
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1-jar/com/nimbusds/jose/jwk/JWKMatcher.java

No license file was found, but licenses were detected in source scan.

/**
 * nimbus-jose-jwt
 *
 * Copyright 2012-2016, Connect2id Ltd.
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use
 * this file except in compliance with the License. You may obtain a copy of the
 * License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software distributed
* under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR
* CONDITIONS OF ANY KIND, either express or implied. See the License for the
* specific language governing permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jwt-8-20-sources-1.jar/com/nimbusds/jose/crypto/AESDecrypter.java
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jwt-8-20-sources-1.jar/com/nimbusds/jose/jwk/source/JWKSource.java
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jwt-8-20-sources-1.jar/com/nimbusds/jose/JWSProvider.java
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jwt-8-20-sources-1.jar/com/nimbusds/jose/jwk/ECPParameterTable.java
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jwt-8-20-sources-1.jar/com/nimbusds/jose/jwk/SecretJWK.java
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jwt-8-20-sources-1.jar/com/nimbusds/jose/proc/JWSVerifierFactory.java
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jwt-8-20-sources-1.jar/com/nimbusds/jose/util/X509CertUtils.java
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jwt-8-20-sources-1.jar/com/nimbusds/jose/KeyException.java
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jwt-8-20-sources-1.jar/com/nimbusds/jose/crypto/DirectEncrypter.java
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jwt-8-20-sources-1.jar/com/nimbusds/jose/proc/JWTProcessorConfiguration.java
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jwt-8-20-sources-1.jar/com/nimbusds/jose/proc/JWEKeySelector.java
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jwt-8-20-sources-1.jar/com/nimbusds/jose/KeyException.java
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jwt-8-20-sources-1.jar/com/nimbusds/jose/package-info.java
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jwt-8-20-sources-1.jar/com/nimbusds/jose/crypto/DefaultEncrypter.java
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jwt-8-20-sources-1.jar/com/nimbusds/jose/proc/JWEEncryptionContext.java
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jwt-8-20-sources-1.jar/com/nimbusds/jose/proc/JWEKeySelector.java
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jwt-8-20-sources-1.jar/com/nimbusds/jose/package-info.java

Open Source Used In security_automation identity-guest-cs 2619
Open Source Used In security_automation identity-guest-cs 2621
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1.jar/com/nimbusds/jose/KeySourceException.java
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1.jar/com/nimbusds/jose/JOSEException.java
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1.jar/com/nimbusds/jose/proc/JWEDecrypterFactory.java
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1.jar/com/nimbusds/jose/proc/DefaultResourceRetriever.java
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1.jar/com/nimbusds/jose/proc/BadJOSEException.java
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1.jar/com/nimbusds/jwt/DefaultResourceRetriever.java
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1.jar/com/nimbusds/jwt/KeyUse.java
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1.jar/com/nimbusds/jwt/JWT.java
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1.jar/com/nimbusds/jwt/JWTParser.java
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1.jar/com/nimbusds/jwt/JWEProvider.java
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1.jar/com/nimbusds/jwt/RSASSASigner.java
* /opt/cola/permits/1136874914_1613969023.23/0/nimbus-jose-jwt-8-20-sources-1.jar/com/nimbusds/jwt/RSASSAVerifier.java

No license file was found, but licenses were detected in source scan.

/
* nimbus-jose-jwt
*
* Copyright 2012-2018, Connect2id Ltd and contributors.
* *
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use
* this file except in compliance with the License. You may obtain a copy of the
* License at
* *
* http://www.apache.org/licenses/LICENSE-2.0
* *
* Unless required by applicable law or agreed to in writing, software distributed
* under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR
* CONDITIONS OF ANY KIND, either express or implied. See the License for the
* specific language governing permissions and limitations under the License.
1.261 kafka-clients 2.7.1

1.261.1 Available under license:

Apache Kafka
Copyright 2021 The Apache Software Foundation.

This product includes software developed at
The Apache Software Foundation (https://www.apache.org/).

This distribution has a binary dependency on jersey, which is available under the CDDL
License. The source code of jersey can be found at https://github.com/jersey/jersey/.

The streams-scala (streams/streams-scala) module was donated by Lightbend and the original code was copyrighted
by them:
Copyright (C) 2018 Lightbend Inc. <https://www.lightbend.com>
Copyright (C) 2017-2018 Alexis Seigneurin.

This project contains the following code copied from Apache Hadoop:
clients/src/main/java/org/apache/kafka/common/utils/PureJavaCrc32C.java
Some portions of this file Copyright (c) 2004-2006 Intel Corporation and licensed under the BSD license.

This project contains the following code copied from Apache Hive:
streams/src/main/java/org/apache/kafka/streams/state/internals/Murmur3.java

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.
2. Grant of Copyright License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
that You distribute, all copyright, patent, trademark, and
attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all}
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.262 spring-javaconfig 1.0.0.m3
1.262.1 Available under license:

No license file was found, but licenses were detected in source scan.
* Copyright 2002-2008 the original author or authors.
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
*   http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1257811092_1643017729.87/0/spring-javaconfig-1-0-0-m3-sources.jar/org/springframework/config/java/util/DefaultScopes.java
* /opt/cola/permits/1257811092_1643017729.87/0/spring-javaconfig-1-0-0-m3-sources.jar/org/springframework/config/java/context/TypeSafeBeanFactory.java
* /opt/cola/permits/1257811092_1643017729.87/0/spring-javaconfig-1-0-0-m3-sources.jar/org/springframework/config/java/annotation/BooleanEnum.java
* /opt/cola/permits/1257811092_1643017729.87/0/spring-javaconfig-1-0-0-m3-sources.jar/org/springframework/config/java/core/ExternalBeanMethodProcessor.java
* /opt/cola/permits/1257811092_1643017729.87/0/spring-javaconfig-1-0-0-m3-sources.jar/org/springframework/config/java/annotation/Configuration.java
* /opt/cola/permits/1257811092_1643017729.87/0/spring-javaconfig-1-0-0-m3-sources.jar/org/springframework/config/java/core/EnhancerMethodInvoker.java
* /opt/ cola/permits/1257811092_1643017729.87/0/spring-javaconfig-1-0-0-m3-sources.jar/org/springframework/config/java/util/TypeSafeBeanFactory.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2002-2008 the original author or authors.
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
*   http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1257811092_1643017729.87/0/spring-javaconfig-1-0-0-m3-sources.jar/org/springframework/config/java/context/TypeSafeBeanFactory.java
* /opt/cola/permits/1257811092_1643017729.87/0/spring-javaconfig-1-0-0-m3-sources.jar/org/springframework/config/java/annotation/BooleanEnum.java
* /opt/cola/permits/1257811092_1643017729.87/0/spring-javaconfig-1-0-0-m3-sources.jar/org/springframework/config/java/core/ExternalBeanMethodProcessor.java
* /opt/cola/permits/1257811092_1643017729.87/0/spring-javaconfig-1-0-0-m3-sources.jar/org/springframework/config/java/annotation/Configuration.java
* /opt/cola/permits/1257811092_1643017729.87/0/spring-javaconfig-1-0-0-m3-sources.jar/org/springframework/config/java/core/EnhancerMethodInvoker.java
* /opt/cola/permits/1257811092_1643017729.87/0/spring-javaconfig-1-0-0-m3-sources.jar/org/springframework/config/java/util/TypeSafeBeanFactory.java
* /opt/cola/permits/1257811092_1643017729.87/0/spring-javaconfig-1-0-0-m3-sources.jar/org/springframework/config/java/context/JavaConfigApplicationContext.java
* /opt/cola/permits/1257811092_1643017729.87/0/spring-javaconfig-1-0-0-m3-sources.jar/org/springframework/config/java/context/AmbiguousBeanLookupException.java
* /opt/cola/permits/1257811092_1643017729.87/0/spring-javaconfig-1-0-0-m3-sources.jar/org/springframework/config/java/core/package-info.java
* /opt/cola/permits/1257811092_1643017729.87/0/spring-javaconfig-1-0-0-m3-sources.jar/org/springframework/config/java/core/AnnotationValueSource.java
* /opt/cola/permits/1257811092_1643017729.87/0/spring-javaconfig-1-0-0-m3-sources.jar/org/springframework/config/java/core/Binder.java
* /opt/cola/permits/1257811092_1643017729.87/0/spring-javaconfig-1-0-0-m3-sources.jar/org/springframework/config/java/core/BeanMethodProcessor.java
* /opt/cola/permits/1257811092_1643017729.87/0/spring-javaconfig-1-0-0-m3-sources.jar/org/springframework/config/java/core/Lazy.java
* /opt/cola/permits/1257811092_1643017729.87/0/spring-javaconfig-1-0-0-m3-sources.jar/org/springframework/config/java/core/LazyInitBean.java
* /opt/cola/permits/1257811092_1643017729.87/0/spring-javaconfig-1-0-0-m3-sources.jar/org/springframework/config/java/core/LazyInitBeanFactory.java
* /opt/cola/permits/1257811092_1643017729.87/0/spring-javaconfig-1-0-0-m3-sources.jar/org/springframework/config/java/core/package-info.java
* /opt/cola/permits/1257811092_1643017729.87/0/spring-javaconfig-1-0-0-m3-sources.jar/org/springframework/config/java/core/BeanFinder.java
* /opt/cola/permits/1257811092_1643017729.87/0/spring-javaconfig-1-0-0-m3-sources.jar/org/springframework/config/java/core/package-info.java
* /opt/cola/permits/1257811092_1643017729.87/0/spring-javaconfig-1-0-0-m3-sources.jar/org/springframework/config/java/core/DefaultServiceLocator.java
1.263 aop-alliance 5.3.12

1.263.1 Available under license:

Apache License
Version 2.0, January 2004
https://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
IMPLIED, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.
Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

=======================================================================

SPRING FRAMEWORK 5.3.12 SUBCOMPONENTS:

Spring Framework 5.3.12 includes a number of subcomponents
with separate copyright notices and license terms. The product that
includes this file does not necessarily use all the open source
subcomponents referred to below. Your use of the source
code for these subcomponents is subject to the terms and
conditions of the following licenses.


Copyright (c) 2000-2011 INRIA, France Telecom
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
   documentation and/or other materials provided with the distribution.

3. Neither the name of the copyright holders nor the names of its
   contributors may be used to endorse or promote products derived from
   this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
THE POSSIBILITY OF SUCH DAMAGE.

Copyright (c) 1999-2009, OW2 Consortium <https://www.ow2.org/>

>>> CGLIB 3.3 (cglib:cglib:3.3):

Per the LICENSE file in the CGLIB JAR distribution downloaded from
https://github.com/cglib/cglib/releases/download/RELEASE_3_3_0/cglib-3.3.0.jar,
CGLIB 3.3 is licensed under the Apache License, version 2.0, the text of which
is included above.

>>> Objenesis 3.1 (org.objenesis:objenesis:3.1):

Per the LICENSE file in the Objenesis ZIP distribution downloaded from
http://objenesis.org/download.html, Objenesis 3.1 is licensed under the
Apache License, version 2.0, the text of which is included above.

Per the NOTICE file in the Objenesis ZIP distribution downloaded from
http://objenesis.org/download.html and corresponding to section 4d of the
Apache License, Version 2.0, in this case for Objenesis:

Objenesis
Copyright 2006-2019 Joe Walnes, Henri Tremblay, Leonardo Mesquita

===============================================================================
To the extent any open source components are licensed under the EPL and/or
other similar licenses that require the source code and/or modifications to
source code to be made available (as would be noted above), you may obtain a
copy of the source code corresponding to the binaries for such open source
components and modifications thereto, if any, (the "Source Files"), by
downloading the Source Files from https://spring.io/projects, Pivotal's website
at https://network.pivotal.io/open-source, or by sending a request, with your
name and address to: Pivotal Software, Inc., 875 Howard Street, 5th floor, San
Francisco, CA 94103, Attention: General Counsel. All such requests should
clearly specify: OPEN SOURCE FILES REQUEST, Attention General Counsel. Pivotal
can mail a copy of the Source Files to you on a CD or equivalent physical medium.

This offer to obtain a copy of the Source Files is valid for three years from the date you acquired this Software product. Alternatively, the Source Files may accompany the Software.

Spring Framework 5.3.12
Copyright (c) 2002-2021 Pivotal, Inc.

This product is licensed to you under the Apache License, Version 2.0 (the "License"). You may not use this product except in compliance with the License.

This product may include a number of subcomponents with separate copyright notices and license terms. Your use of the source code for these subcomponents is subject to the terms and conditions of the subcomponent's license, as noted in the license.txt file.

1.264 zookeeper 3.5.5
1.264.1 Available under license :

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.
"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual, 
worldwide, non-exclusive, no-charge, royalty-free, irrevocable 
(except as stated in this section) patent license to make, have made, 
use, offer to sell, sell, import, and otherwise transfer the Work, 
where such license applies only to those patent claims licensable 
by such Contributor that are necessarily infringed by their 
Contribution(s) alone or by combination of their Contribution(s) 
with the Work to which such Contribution(s) was submitted. If You 
institute patent litigation against any entity (including a 
cross-claim or counterclaim in a lawsuit) alleging that the Work 
or a Contribution incorporated within the Work constitutes direct 
or contributory patent infringement, then any patent licenses 
granted to You under this License for that Work shall terminate 
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the 
Work or Derivative Works thereof in any medium, with or without 
modifications, and in Source or Object form, provided that You 
meet the following conditions:

(a) You must give any other recipients of the Work or 
    Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices 
stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works 
    that You distribute, all copyright, patent, trademark, and 
attribution notices from the Source form of the Work, 
    excluding those notices that do not pertain to any part of 
    the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its 
distribution, then any Derivative Works that You distribute must 
include a readable copy of the attribution notices contained 
within such NOTICE file, excluding those notices that do not 
pertain to any part of the Derivative Works, in at least one 
of the following places: within a NOTICE text file distributed 
as part of the Derivative Works; within the Source form or 
documentation, if provided along with the Derivative Works; or, 
within a display generated by the Derivative Works, if and 
wherever such third-party notices normally appear. The contents 
of the NOTICE file are for informational purposes only and 
do not modify the License. You may add Your own attribution 
notices within Derivative Works that You distribute, alongside 
or as an addendum to the NOTICE text from the Work, provided 
that such additional attribution notices cannot be construed 
as modifying the License.
You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.265 annotations 20.1.0
1.265.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2000-2020 JetBrains s.r.o.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * *
 * https://www.apache.org/licenses/LICENSE-2.0
 * */
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1103213823_1604039616.52/0/annotations-20-1-0-sources-jar/org/jetbrains/annotations/Contract.java
* /opt/cola/permits/1103213823_1604039616.52/0/annotations-20-1-0-sources-jar/org/jetbrains/annotations/ApiStatus.java
* /opt/cola/permits/1103213823_1604039616.52/0/annotations-20-1-0-sources-jar/org/jetbrains/annotations/Subst.java
* /opt/cola/permits/1103213823_1604039616.52/0/annotations-20-1-0-sources-jar/org/jetbrains/annotations/NonNls.java
* /opt/cola/permits/1103213823_1604039616.52/0/annotations-20-1-0-sources-jar/org/jetbrains/annotations/Range.java
* /opt/cola/permits/1103213823_1604039616.52/0/annotations-20-1-0-sources-jar/org/jetbrains/annotations/PropertyKey.java
* /opt/cola/permits/1103213823_1604039616.52/0/annotations-20-1-0-sources-jar/org/jetbrains/annotations/NotNull.java
* /opt/cola/permits/1103213823_1604039616.52/0/annotations-20-1-0-sources-jar/org/intellij/lang/annotations/MagicConstant.java
* /opt/cola/permits/1103213823_1604039616.52/0/annotations-20-1-0-sources-jar/org/intellij/lang/annotations/RegExp.java
* /opt/cola/permits/1103213823_1604039616.52/0/annotations-20-1-0-sources-jar/org/jetbrains/annotations/Async.java
* /opt/cola/permits/1103213823_1604039616.52/0/annotations-20-1-0-sources-jar/org/jetbrains/annotations/TestOnly.java
* /opt/cola/permits/1103213823_1604039616.52/0/annotations-20-1-0-sources-jar/org/jetbrains/annotations/MustBeInvokedByOverriders.java
* /opt/cola/permits/1103213823_1604039616.52/0/annotations-20-1-0-sources-jar/org/intellij/lang/annotations/Flow.java
* /opt/cola/permits/1103213823_1604039616.52/0/annotations-20-1-0-sources-jar/org/intellij/lang/annotations/Identifier.java
* /opt/cola/permits/1103213823_1604039616.52/0/annotations-20-1-0-sources-jar/org/intellij/lang/annotations/Language.java
* /opt/cola/permits/1103213823_1604039616.52/0/annotations-20-1-0-sources-jar/org/jetbrains/annotations/UnmodifiableView.java
* /opt/cola/permits/1103213823_1604039616.52/0/annotations-20-1-0-sources-jar/org/jetbrains/annotations/Unmodifiable.java
* /opt/cola/permits/1103213823_1604039616.52/0/annotations-20-1-0-sources-
1.266 hdrhistogram 2.1.12

1.266.1 Available under license:

No license file was found, but licenses were detected in source scan.

/**
 * Written by Gil Tene of Azul Systems, and released to the public domain,
 * as explained at http://creativecommons.org/publicdomain/zero/1.0/
 */

Found in path(s):
* /opt/cola/permits/1257203437_1642787675.09/0(hdrhistogram-2-1-12-sources-jar/org/HdrHistogram/DoublePercentileIterator.java
* /opt/cola/permits/1257203437_1642787675.09/0(hdrhistogram-2-1-12-sources-jar/org/HdrHistogram/ZigZagEncoding.java
* /opt/cola/permits/1257203437_1642787675.09/0(hdrhistogram-2-1-12-sources-jar/org/HdrHistogram/AllValuesIterator.java
* /opt/cola/permits/1257203437_1642787675.09/0(hdrhistogram-2-1-12-sources-jar/org/HdrHistogram/Base64Helper.java
* /opt/cola/permits/1257203437_1642787675.09/0(hdrhistogram-2-1-12-sources-jar/org/HdrHistogram/IntCountsHistogram.java
* /opt/cola/permits/1257203437_1642787675.09/0(hdrhistogram-2-1-12-sources-jar/org/HdrHistogram/DoubleLinearIterator.java
* /opt/cola/permits/1257203437_1642787675.09/0(hdrhistogram-2-1-12-sources-jar/org/HdrHistogram/PackedConcurrentDoubleHistogram.java
* /opt/cola/permits/1257203437_1642787675.09/0(hdrhistogram-2-1-12-sources-jar/org/HdrHistogram/SynchronizedHistogram.java
jar/org/HdrHistogram/DoubleLogarithmicIterator.java
* /opt/cola/permits/1257203437_1642787675.09/0/hdrhistogram-2-1-12-sources-jar/org/HdrHistogram/ShortCountsHistogram.java
* /opt/cola/permits/1257203437_1642787675.09/0/hdrhistogram-2-1-12-sources-jar/org/HdrHistogram/EncodableHistogram.java
* /opt/cola/permits/1257203437_1642787675.09/0/hdrhistogram-2-1-12-sources-jar/org/HdrHistogram/HistogramLogScanner.java
* /opt/cola/permits/1257203437_1642787675.09/0/hdrhistogram-2-1-12-sources-jar/org/HdrHistogram/packedarray/PackedArrayRecorder.java
* /opt/cola/permits/1257203437_1642787675.09/0/hdrhistogram-2-1-12-sources-jar/org/HdrHistogram/ConcurrentHistogram.java

No license file was found, but licenses were detected in source scan.

/*
 * Written by Gil Tene of Azul Systems, and released to the public domain,
 * as explained at http://creativecommons.org/publicdomain/zero/1.0/
 *
 * @author Gil Tene
 */

Found in path(s):
* /opt/cola/permits/1257203437_1642787675.09/0/hdrhistogram-2-1-12-sources-jar/org/HdrHistogram/AbstractHistogram.java
* /opt/cola/permits/1257203437_1642787675.09/0/hdrhistogram-2-1-12-sources-jar/org/HdrHistogram/DoubleHistogram.java

No license file was found, but licenses were detected in source scan.

/*
 * package-info.java
 * Written by Gil Tene of Azul Systems, and released to the public domain,
 * as explained at http://creativecommons.org/publicdomain/zero/1.0/
 */

Found in path(s):
* /opt/cola/permits/1257203437_1642787675.09/0/hdrhistogram-2-1-12-sources-jar/org/HdrHistogram/package-info.java

No license file was found, but licenses were detected in source scan.

/**
 * Written by Gil Tene of Azul Systems, and released to the public domain,
 * as explained at http://creativecommons.org/publicdomain/zero/1.0/
 *
 * @author Gil Tene
 */

package org.HdrHistogram;

final class Version {
public static final String version="$VERSION$";
public static final String build_time="$BUILD_TIME$";
}

1.267 commons-beanutils 1.9.4

1.267.1 Available under license:

Apache Commons BeanUtils
Copyright 2000-2019 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
   Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
   stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
   that You distribute, all copyright, patent, trademark, and
   attribution notices from the Source form of the Work,
   excluding those notices that do not pertain to any part of
   the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.
APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.268 commons-validator 1.6

1.268.1 Available under license :

Apache Commons Validator
Copyright 2001-2017 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

    Apache License
    Version 2.0, January 2004
    http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.
"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
1.269 spring-security-rsa 1.0.10.RELEASE

1.269.1 Available under license:
No license file was found, but licenses were detected in source scan.

<url>http://www.apache.org/licenses/LICENSE-2.0.txt</url>

Found in path(s):

1.270 reflections 0.9.11

1.270.1 Available under license:
No license file was found, but licenses were detected in source scan.

<name>The New BSD License</name>

Found in path(s):
* /opt/cola/permits/1000000666_1646171445.29/0/reflections-0-9-11-sources-jar/META-INF/maven/org.reflections/reflections/pom.xml

1.271 aws-java-sdk-core 1.11.415

1.271.1 Available under license:
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2012-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 * Licensed under the Apache License, Version 2.0 (the "License").
 * You may not use this file except in compliance with the License.
 * A copy of the License is located at
 * http://aws.amazon.com/apache2.0
 *
 * or in the "license" file accompanying this file. This file is distributed
 * on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing
 * permissions and limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-
jar/com/amazonaws/internal/CustomBackoffStrategy.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/internal/EC2MetadataClient.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/http/IdleConnectionReaper.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/HandlerContextAware.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/auth/DefaultAWSCredentialsProviderChain.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/internal/DynamoDBBackoffStrategy.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/auth/EnvironmentVariableCredentialsProvider.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/auth/InstanceProfileCredentialsProvider.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/auth/AWSCredentialsProviderChain.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/internal/CRC32MismatchException.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/auth/SystemPropertiesCredentialsProvider.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/auth/ClasspathPropertiesFileCredentialsProvider.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/util/AwsHostNameUtils.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2012-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 * 
 * Licensed under the Apache License, Version 2.0 (the "License").
 * You may not use this file except in compliance with the License.
 * A copy of the License is located at
 * 
 * http://aws.amazon.com/apache2.0
 * 
 * or in the "license" file accompanying this file. This file is distributed
 * on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing
 * permissions and limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/handlers/AsyncHandler.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2010-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
*Licensed under the Apache License, Version 2.0 (the "License").
*You may not use this file except in compliance with the License.
*A copy of the License is located at
*http://aws.amazon.com/apache2.0
*
or in the "license" file accompanying this file. This file is distributed
*on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
*express or implied. See the License for the specific language governing
*permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/internal/config/JsonIndex.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/Protocol.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/internal/config/SignerConfigJsonHelper.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/util/ImmutableMapParameter.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/internal/config/SignerConfig.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/internal/config/Builder.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2012-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License"). You may not use this file except in compliance with
* the License. A copy of the License is located at
*http://aws.amazon.com/apache2.0
*
or in the "license" file accompanying this file. This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR
* CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions
* and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/annotation/GuardedBy.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2011-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 *
 * Licensed under the Apache License, Version 2.0 (the "License").
 * You may not use this file except in compliance with the License.
 * A copy of the License is located at
 *
 * http://aws.amazon.com/apache2.0
 *
 * or in the "license" file accompanying this file. This file is distributed
 * on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing
 * permissions and limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-
  jar/com/amazonaws/protocol/json/internal/JsonProtocolMarshaller.java

No license file was found, but licenses were detected in source scan.

/**
 * Copyright 2016-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 * <p/>
 * Licensed under the Apache License, Version 2.0 (the "License"). You may not use this file except in compliance
 * with
 * the License. A copy of the License is located at
 * <p/>
 * http://aws.amazon.com/apache2.0
 * <p/>
 * or in the "license" file accompanying this file. This file is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR
 * CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing
permissions
* and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/partitions/model/Partitions.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2014-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
* Portions copyright 2006-2009 James Murty. Please see LICENSE.txt
* for applicable license terms and NOTICE.txt for applicable notices.
* Licensed under the Apache License, Version 2.0 (the "License").
* You may not use this file except in compliance with the License.
* A copy of the License is located at
* http://aws.amazon.com/apache2.0
*
* or in the "license" file accompanying this file. This file is distributed
* on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express or implied. See the License for the specific language governing
* permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/internal/ResettableInputStream.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/internal/ReleasableInputStream.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2011-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at:
* http://aws.amazon.com/apache2.0
*
* This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES
* OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and
* limitations under the License.
*/
Found in path(s):
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1.jar/com/amazonaws/internal/ConnectionUtils.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1.jar/com/amazonaws/internal/EC2CredentialsUtils.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1.jar/com/amazonaws/auth/EC2CredentialsFetcher.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1.jar/com/amazonaws/auth/EC2ContainerCredentialsProviderWrapper.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1.jar/com/amazonaws/auth/ContainerCredentialsProvider.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1.jar/com/amazonaws/auth/ContainerCredentialsRetryPolicy.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1.jar/com/amazonaws/auth/EndpointPrefixAwareSigner.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1.jar/com/amazonaws/retry/internal/CredentialsEndpointRetryPolicy.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1.jar/com/amazonaws/retry/internal/CredentialsEndpointRetryPolicy.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1.jar/com/amazonaws/retry/internal/CredentialsEndpointRetryParameters.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1.jar/com/amazonaws/retry/internal/CredentialsEndpointRetryParameters.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1.jar/com/amazonaws/auth/EndpointPrefixAwareSigner.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1.jar/com/amazonaws/Auth/EC2ContainerCredentialsProviderWrapper.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015-2018 Amazon Technologies, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at:
 * *
 * http://aws.amazon.com/apache2.0
 *
 * This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES
 * OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1.jar/com/amazonaws/internal/SdkSSLContext.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1.jar/com/amazonaws/annotation/package-info.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2016-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 */
Licensed under the Apache License, Version 2.0 (the "License").
You may not use this file except in compliance with the License.
A copy of the License is located at
http://aws.amazon.com/apache2.0
or in the "license" file accompanying this file. This file is distributed
on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
express or implied. See the License for the specific language governing
permissions and limitations under the License.

Found in path(s):
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/http/apache/SdkProxyRoutePlanner.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/partitions/PartitionsLoader.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/regions/RegionMetadataFactory.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/http/apache/client/impl/CRC32ChecksumResponseInterceptor.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/regions/RegionImpl.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/protocol/json/SdkIonGenerator.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/regions/RegionMetadataProvider.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/partitions/model/CredentialScope.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/regions/InMemoryRegionImpl.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/regions/LegacyRegionXmlLoadUtils.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/regions/AbstractRegionMetadataProvider.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/partitions/model/Endpoint.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/partitions/PartitionRegionImpl.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/partitions/model/Service.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/regions/LegacyRegionXmlMetadataBuilder.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/partitions/model/Region.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/partitions/PartitionMetadataProvider.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2013-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 *
 *Licensed under the Apache License, Version 2.0 (the "License").
 *You may not use this file except in compliance with the License.
 *A copy of the License is located at
 *
 *http://aws.amazon.com/apache2.0
 *
 *or in the "license" file accompanying this file. This file is distributed
 *on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 *express or implied. See the License for the specific language governing
 *permissions and limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/util/Base32.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/auth/AWS4Signer.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/util/CodecUtils.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/util/Base32Codec.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/util/Base16Codec.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/internal/MetricAware.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/util/Throwables.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/internal/SdkDigestInputStream.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/util/EC2MetadataUtils.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/util/Codec.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/partitions/model/Partition.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/transform/SimpleTypeCborUnmarshallers.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/regions/InMemoryRegionsProvider.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/protocol/json/SdkStructuredIonFactory.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/util/Base16.java
No license file was found, but licenses were detected in source scan.
/*
 * Copyright 2013-2018 Amazon Technologies, Inc.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at:
 * http://aws.amazon.com/apache2.0
 *
 * This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES
 * OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/regions/Regions.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/regions/ServiceAbbreviations.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 * Licensed under the Apache License, Version 2.0 (the "License").
 * You may not use this file except in compliance with the License.
 * A copy of the License is located at
 * http://aws.amazon.com/apache2.0
 *
 * or in the "license" file accompanying this file. This file is distributed
 * on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing
 * permissions and limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/internal/SdkIOUtils.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/ReadLimitInfo.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/internal/JsonErrorMessageParser.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/util/ValidationUtils.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/PredefinedClientConfigurations.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/internal/DelegateInputStream.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/http/timers/client/ClientExecutionAbortTaskImpl.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/http/timers/client/ClientExecutionAbortTrackerTask.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/http/timers/client/ClientExecutionTimeoutException.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/log/CommonsLogFactory.java
* /opt/ cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/log/JulLog.java

Open Source Used In security_automation identity-guest-cs 2668
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2011-2018 Amazon Technologies, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at:
 *
 *    http://aws.amazon.com/apache2.0
 *
 * This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES
 * OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s): 
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/http/timers/request/HttpRequestAbortTask.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/internal/http/JsonErrorCodeParser.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/util/json/Jackson.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/auth/BasicSessionCredentials.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/RequestClientOptions.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/jmx/spi/JmxInfoProvider.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/auth/AWSSessionCredentialsProvider.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/jmx/JmxInfoProviderSupport.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/auth/AWSSessionCredentials.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/util/ClassLoaderHelper.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/metrics/MetricAdmin.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/metrics/MetricAdminMBean.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/util/ClassLoaderHelper.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/metrics/MetricAdmin.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/util/ClassLoaderHelper.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/metrics/MetricAdmin.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/util/ClassLoaderHelper.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/metrics/MetricAdmin.java

No license file was found, but licenses were detected in source scan.
/*
 * Licensed to the Apache Software Foundation (ASF) under one
 * or more contributor license agreements. See the NOTICE file
 * distributed with this work for additional information
 * regarding copyright ownership. The ASF licenses this file
 * to you under the Apache License, Version 2.0 (the
 * "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at
 */

* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing,
* software distributed under the License is distributed on an
* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
* KIND, either express or implied. See the License for the
* specific language governing permissions and limitations
* under the License.
*
* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
*
*/

Found in path(s):
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-
  jar/com/amazonaws/util/BasicNameValuePair.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-
  jar/com/amazonaws/util/NameValuePair.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-
  jar/com/amazonaws/util/URLEncodedUtils.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2016. Amazon.com, Inc. or its affiliates. All Rights Reserved.
 *
 * Licensed under the Apache License, Version 2.0 (the "License").
 * You may not use this file except in compliance with the License.
 * A copy of the License is located at
 */

* http://aws.amazon.com/apache2.0
*
* or in the "license" file accompanying this file. This file is distributed
* on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express or implied. See the License for the specific language governing
* permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1.jar/com/amazonaws/util/IdempotentUtils.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1.jar/com/amazonaws/auth/CanHandleNullCredentials.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1.jar/com/amazonaws/http/apache/client/impl/ApacheConnectionManagerFactory.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1.jar/com/amazonaws/http/apache/impl/ApacheHttpRequestFactory.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1.jar/com/amazonaws/http/apache/client/impl/ApacheHttpClientFactory.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1.jar/com/amazonaws/log/InternalLogFactory.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1.jar/com/amazonaws/handlers/IRequestHandler2.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1.jar/com/amazonaws/http/apache/client/impl/ConnectionManagerAwareHttpClient.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1.jar/com/amazonaws/internal/DateTimeJsonSerializer.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1.jar/com/amazonaws/http/timers/package-info.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1.jar/com/amazonaws/util/FakeIOException.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1.jar/com/amazonaws/http/conn/ssl/ShouldClearSslSessionPredicate.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1.jar/com/amazonaws/http/apache/utils/ApacheUtils.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1.jar/com/amazonaws/http/ssl/ShouldClearSslSessionPredicate.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1.jar/com/amazonaws/http/apache/client/impl/HttpHost.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2011-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 *
 * Licensed under the Apache License, Version 2.0 (the "License").
 * You may not use this file except in compliance with the License.
 * A copy of the License is located at
 *
 * http://aws.amazon.com/apache2.0
 *
 * or in the "license" file accompanying this file. This file is distributed
* on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express or implied. See the License for the specific language governing
* permissions and limitations under the License.
*/
/**
* Interface for providing AWS credentials. Implementations are free to use any
* strategy for providing AWS credentials, such as simply providing static
* credentials that don't change, or more complicated implementations, such as
* integrating with existing key management systems.
*/

Found in path(s):
*/opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-
jar/com/amazonaws/auth/AWSCredentialsProvider.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2011-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License").
* You may not use this file except in compliance with the License.
* A copy of the License is located at
*
* http://aws.amazon.com/apache2.0
*
* or in the "license" file accompanying this file. This file is distributed
* on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express or implied. See the License for the specific language governing
* permissions and limitations under the License.
*/
/**
* Interface for providing AWS region information. Implementations are free to use any strategy for
* providing region information.
*/

Found in path(s):
*/opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-
jar/com/amazonaws/regions/AwsRegionProvider.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2014-2018 Amazon Technologies, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at:
*
* http://aws.amazon.com/apache2.0
* or in the "license" file accompanying this file. This file is distributed
* on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express or implied. See the License for the specific language governing
* permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-
  jar/com/amazonaws/auth/profile/ProfilesConfigFileWriter.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-
  jar/com/amazonaws/internal/BoundedLinkedListMap.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-
  jar/com/amazonaws/auth/profile/ProfilesConfigFile.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-
  jar/com/amazonaws/internal/SignerKey.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-
  jar/com/amazonaws/auth/profile/ProfilesConfigFile.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-
  jar/com/amazonaws/auth/internal/Releasable.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-
  jar/com/amazonaws/internal/SdkInputStream.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-
  jar/com/amazonaws/auth/profile/internal/AbstractProfilesConfigFileScanner.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-
  jar/com/amazonaws/event/DeliveryMode.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-
  jar/com/amazonaws/auth/profile/internal/securitytoken/STSProfileCredentialsServiceProvider.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-
  jar/com/amazonaws/event/SDKProgressPublisher.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-
  jar/com/amazonaws/event/RequestProgressInputStream.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-
  jar/com/amazonaws/event/ResponseProgressInputStream.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-
  jar/com/amazonaws/event/ProgressEventFilter.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-
  jar/com/amazonaws/event/ProgressTracker.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-
  jar/com/amazonaws/http/UnreliableTestConfig.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-
  jar/com/amazonaws/event/ProgressInputStream.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-
  jar/com/amazonaws/internal/SdkSSLSocket.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-
Found in path(s):
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-
  jar/com/amazonaws/http/JsonResponseHandler.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-
  jar/com/amazonaws/protocol/json/SdkCborGenerator.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-
  jar/com/amazonaws/protocol/json/SdkStructuredPlainJsonFactory.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-
  jar/com/amazonaws/protocol/json/SdkStructuredCborFactory.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-
  jar/com/amazonaws/protocol/json/SdkStructuredCborFactory.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (c) 2016 Amazon.com, Inc. or its affiliates. All Rights Reserved.
* 
* Licensed under the Apache License, Version 2.0 (the "License").
* You may not use this file except in compliance with the License.
* A copy of the License is located at
* 
* http://aws.amazon.com/apache2.0
* 
* or in the "license" file accompanying this file. This file is distributed
* on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express or implied. See the License for the specific language governing
* permissions and limitations under the License.
*
*/
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2016-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 *
 * Licensed under the Apache License, Version 2.0 (the "License"). You may not
 * use this file except in compliance with the License. A copy of the License is
 * located at
 *
 * http://aws.amazon.com/apache2.0
 *
 * or in the "license" file accompanying this file. This file is distributed on
 * an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing
 * permissions and limitations under the License.
 */

Found in path(s):

/*
 * Copyright 2012-2018 Amazon Technologies, Inc.
 *
 * Portions copyright 2006-2009 James Murty. Please see LICENSE.txt
 * for applicable license terms and NOTICE.txt for applicable notices.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at:
 *
 * http://aws.amazon.com/apache2.0
 *
 * This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES
 * OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2018-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 * Licensed under the Apache License, Version 2.0 (the "License").
 * You may not use this file except in compliance with the License.
 * A copy of the License is located at
 * http://aws.amazon.com/apache2.0
 * or in the "license" file accompanying this file. This file is distributed
 * on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing
 * permissions and limitations under the License.
 */

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 * Licensed under the Apache License, Version 2.0 (the "License").
 * You may not use this file except in compliance with the License.
 * A copy of the License is located at
 * http://aws.amazon.com/apache2.0
 * or in the "license" file accompanying this file. This file is distributed
 * on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing
 * permissions and limitations under the License.
 */

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2010-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 */
Licensed under the Apache License, Version 2.0 (the "License").
You may not use this file except in compliance with the License.
A copy of the License is located at

http://aws.amazon.com/apache2.0

or in the "license" file accompanying this file. This file is distributed
on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
express or implied. See the License for the specific language governing
permissions and limitations under the License.

*Licensed under the Apache License, Version 2.0 (the "License").
You may not use this file except in compliance with the License.
A copy of the License is located at

http://aws.amazon.com/apache2.0

or in the "license" file accompanying this file. This file is distributed
on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
express or implied. See the License for the specific language governing
permissions and limitations under the License.
*/

Found in path(s):
*/opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-
jar/com/amazonaws/auth/policy/Principal.java
*/opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-
jar/com/amazonaws/waiters/WaiterExecutionBuilder.java
*/opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-
jar/com/amazonaws/metrics/AwsSdkMetrics.java
*/opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-
jar/com/amazonaws/retry/PredefinedRetryPolicies.java
*/opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-
jar/com/amazonaws/monitoring/CsmConfiguration.java
*/opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-
jar/com/amazonaws/util/AwsClientSideMonitoringMetrics.java
*/opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-
jar/com/amazonaws/auth/Signer.java
*/opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-
jar/com/amazonaws/http/impl/client/SdkHttpRequestRetryHandler.java
*/opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-
jar/com/amazonaws/handlers/HandlerAfterAttemptContext.java
*/opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-
jar/com/amazonaws/auth/Signer.java
*/opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-
jar/com/amazonaws/http/HttpRequest.java
*/opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-
jar/com/amazonaws/http/HttpResponse.java
*/opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-
jar/com/amazonaws/http/apache/request/impl/HttpGetWithBody.java
*/opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-
jar/com/amazonaws/util/AWSRequestMetricsFullSupport.java
*/opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-
jar/com/amazonaws/waiters/WaiterHandler.java
*/opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-
jar/com/amazonaws/auth/policy/conditions/NumericCondition.java
*/opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-
jar/com/amazonaws/endpointdiscovery/DefaultEndpointDiscoveryProviderChain.java
Open Source Used In security_automation identity-guest-cs 2681

* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1.jar/com/amazonaws/monitoring/MonitoringListener.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1.jar/com/amazonaws/auth/policy/resources/package-info.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1.jar/com/amazonaws/monitoring/CsmConfigurationProvider.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1.jar/com/amazonaws/auth/BasicAWSCredentials.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1.jar/com/amazonaws/interfaces/RequestHandler.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1.jar/com/amazonaws/interfaces/RequestHandler2.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1.jar/com/amazonaws/core/ProxyAuthenticationMethod.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1.jar/com/amazonaws/transform/MapEntry.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1.jar/com/amazonaws/transform/SimpleTypeStaxUnmarshallers.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1.jar/com/amazonaws/auth/policy/conditions/IpAddressCondition.java
Open Source Used In security_automation identity-guest-cs 2684

* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/metrics/internal/ServiceMetricTypeGuesser.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/endpointdiscovery/EndpointDiscoveryProvider.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/auth/SignerFactory.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/cache/Cache.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/metrics/MetricFilterInputStream.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/monitoring/internal/AuthRetryParameters.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/auth/policy/conditions/package-info.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/monitoring/DefaultCsmConfigurationProviderChain.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/metrics/ServiceMetricType.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/transform/StandardErrorUnmarshaller.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/metrics/ByteThroughputHelper.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/waiters/PollingStrategy.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/internal/config/HostRegexToRegionMapping.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/metrics/ThroughputMetricType.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/cache/CacheLoader.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/waiters/FixedDelayStrategy.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/event/ProgressListener.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/transform/SimpleTypeJsonUnmarshallers.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/util/AWSServiceMetrics.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/monitoring/MonitoringEvent.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/transform/VoidStaxUnmarshaller.java
No license file was found, but licenses were detected in source scan.

/*
* Licensed to the Apache Software Foundation (ASF) under one or more

* contributor license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright ownership.
* The ASF licenses this file to You under the Apache License, Version 2.0
* (the "License"); you may not use this file except in compliance with
* the License. You may obtain a copy of the License at
* *
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-
jar/com/amazonaws/log/InternalLogApi.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2011-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
* *
* Licensed under the Apache License, Version 2.0 (the "License"). You may not
* use this file except in compliance with the License. A copy of the License is
* located at
* *
* http://aws.amazon.com/apache2.0
*
* or in the "license" file accompanying this file. This file is distributed on
* an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express or implied. See the License for the specific language governing
* permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-
jar/com/amazonaws/protocol/json/IonFactory.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-
jar/com/amazonaws/protocol/json/IonParser.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-
jar/com/amazonaws/transform/SimpleTypeIonUnmarshallers.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2015-2018 Amazon.com, Inc. or its affiliates. All Rights
* Reserved.
* */
Licensed under the Apache License, Version 2.0 (the "License").
You may not use this file except in compliance with the License.
A copy of the License is located at
* http://aws.amazon.com/apache2.0
*
or in the "license" file accompanying this file. This file is
distributed
on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
KIND, either
express or implied. See the License for the specific language
governing
permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1.jar/com/amazonaws/util/CredentialUtils.java
No license file was found, but licenses were detected in source scan.

*/

Copyright 2010-2018 Amazon.com, Inc. or its affiliates. All Rights
Reserved.

Licensed under the Apache License, Version 2.0 (the "License").
You may not use this file except in compliance with the License.
A copy of the License is located at
* http://aws.amazon.com/apache2.0
*
or in the "license" file accompanying this file. This file is distributed
on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
express or implied. See the License for the specific language governing
permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1.jar/com/amazonaws/internal/auth/NoOpSignerProvider.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1.jar/com/amazonaws/adapters/types/TypeAdapter.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1.jar/com/amazonaws/util/NullResponseMetadataCache.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1.jar/com/amazonaws/internal/auth/SignerProvider.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1.jar/com/amazonaws/adapters/types/StringToInputStreamAdapter.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-
/*
 * Copyright 2011-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 * Licensed under the Apache License, Version 2.0 (the "License").
 * You may not use this file except in compliance with the License.
 * A copy of the License is located at
 * http://aws.amazon.com/apache2.0
 * or in the "license" file accompanying this file. This file is distributed
 * on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing
 * permissions and limitations under the License.
 */

No license file was found, but licenses were detected in source scan.
jar/com/amazonaws/profile/path/config/SharedConfigDefaultLocationProvider.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/retry/v2/AndRetryCondition.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/retry/v2/SimpleRetryPolicy.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/retry/v2/OrRetryPolicy.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/regions/AwsSystemPropertyRegionProvider.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/protocol/json/JsonContentTypeResolver.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/protocol/MarshallingInfo.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/protocol/MarshallingLocation.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/util/StringInputStream.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/util/XMLWriter.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/util/XMLWriter.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/util/TimingInfo.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/client/AwsAsyncClientParams.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/util/XMLWriter.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/client/AwsAsyncClientParams.java
* /opt/cola/permits/1136061426_1613670290.05/0/aws-java-sdk-core-1-11-415-sources-1-jar/com/amazonaws/util/XMLWriter.java
/*
   Copyright 2010-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.

   Portions copyright 2006-2009 James Murty. Please see LICENSE.txt
   for applicable license terms and NOTICE.txt for applicable notices.
*/

No license file was found, but licenses were detected in source scan.

Open Source Used In security_automation identity-guest-cs 2694
1.272 jackson 2.13.0

1.272.1 Available under license:

# Jackson JSON processor

Jackson is a high-performance, Free/Open Source JSON processing library. It was originally written by Tatu Saloranta (tatu.saloranta@iki.fi), and has been in development since 2007. It is currently developed by a community of developers.

## Licensing

Jackson 2.x core and extension components are licensed under Apache License 2.0. To find the details that apply to this artifact see the accompanying LICENSE file.

## Credits

A list of contributors may be found from CREDITS(-2.x) file, which is included in some artifacts (usually source distributions); but is always available from the source code management (SCM) system project uses.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION
1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted"
"means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and
attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.273 activation-api 1.2.2
1.273.1 Available under license:

Copyright (c) 2018 Oracle and/or its affiliates. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

- Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.

- Redistributions in binary form must reproduce the above copyright
  notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

- Neither the name of the Eclipse Foundation, Inc. nor the names of its
  contributors may be used to endorse or promote products derived
  from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS
IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO,
THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR
CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR
PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF
LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

# Notices for Jakarta Activation

This content is produced and maintained by Jakarta Activation project.

* Project home: https://projects.eclipse.org/projects/ee4j.jaf

## Copyright

All content is the property of the respective authors or their employers. For
more information regarding authorship of content, please consult the listed
source code repository logs.

## Declared Project Licenses

This program and the accompanying materials are made available under the terms of the Eclipse Distribution License v. 1.0, which is available at http://www.eclipse.org/org/documents/edl-v10.php.

SPDX-License-Identifier: BSD-3-Clause

## Source Code

The project maintains the following source code repositories:

* https://github.com/eclipse-ee4j/jaf

## Third-party Content

This project leverages the following third party content.

JUnit (4.12)

* License: Eclipse Public License

### 1.274 spring-cloud-context 3.0.4

### 1.274.1 Available under license:

No license file was found, but licenses were detected in source scan.

<name>Apache License, Version 2.0</name>
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at
distributed under the License is distributed on an "AS IS" BASIS,
Found in path(s):
* /opt/cola/permits/1222421586_1635579139.26/0/spring-cloud-context-3-0-4-jar/META-INF/maven/org.springframework.cloud/spring-cloud-context/pom.xml

### 1.275 converter-moshi 2.9.0

### 1.275.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2015 Square, Inc.
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1125278247_1611280592.09/0/converter-moshi-2-9-0-sources-jar/retrofit2/converter/moshi/MoshiResponseBodyConverter.java
* /opt/cola/permits/1125278247_1611280592.09/0/converter-moshi-2-9-0-sources-jar/retrofit2/converter/moshi/MoshiConverterFactory.java
* /opt/cola/permits/1125278247_1611280592.09/0/converter-moshi-2-9-0-sources-jar/retrofit2/converter/moshi/MoshiRequestBodyConverter.java

1.276 oro 2.0.7

1.276.1 Available under license :

Daniel Savarese <dfs at savarese.org> is the original author of the
OROMatcher, PerlTools, AwkTools, and TextTools packages that became
the Jakarta-ORO project.

Jon Stevens <jon at collab.net> helped prepare the first release of
jakarta-oro and is a constant help in keeping the project consistent
with the Jakarta project as a whole.

Takashi Okamoto <tokamoto at rd.nttdata.co.jp> has contributed a unicode
class character fix and an initial posix character class implementation.

Mark Murphy <markm at tyrell.com> has contributed performance
improvements to Perl5Substitution as well as adding support for
\u\L\uE and escaping of $.

Michael Davey <michael.davey at coderage.org> fixed some documentation and
added a missing int substitute(...) method to Perl5Util.

Harald Kuhn <harald at ontopia.net> updated
MatchActionProcessor.processMatches() to accommodate character encodings.
/* ====================================================================
 * The Apache Software License, Version 1.1
 *
 * Copyright (c) 2000-2002 The Apache Software Foundation. All rights
 * reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 *
 * 1. Redistributions of source code must retain the above copyright
 *    notice, this list of conditions and the following disclaimer.
 *
 * 2. Redistributions in binary form must reproduce the above copyright
 *    notice, this list of conditions and the following disclaimer in
 *    the documentation and/or other materials provided with the
 *    distribution.
 *
 * 3. The end-user documentation included with the redistribution,
 *    if any, must include the following acknowledgment:
 *    "This product includes software developed by the
 *    Apache Software Foundation (http://www.apache.org/)."
 *    Alternately, this acknowledgment may appear in the software itself,
 *    if and wherever such third-party acknowledgments normally appear.
 *
 * 4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
 *    must not be used to endorse or promote products derived from this
 *    software without prior written permission. For written
 *    permission, please contact apache@apache.org.
 *
 * 5. Products derived from this software may not be called "Apache"
 *    or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
 *    name, without prior written permission of the Apache Software Foundation.
 *
 * THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
 * WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
 * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
 * DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
 * ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
 * SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
 * LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
 * USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
 * ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
 * OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
 * OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
 * SUCH DAMAGE.
 * ====================================================================
 */
* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
* [http://www.apache.org/].
*
* Portions of this software are based upon software originally written
* by Daniel F. Savarese. We appreciate his contributions.
*/

1.277 kafka-avro-serializer 5.3.0

1.277.1 Available under license:

No license file was found, but licenses were detected in source scan.

```xml
<project xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns="http://maven.apache.org/POM/4.0.0"
    xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/maven-v4_0_0.xsd">

  <modelVersion>4.0.0</modelVersion>

  <parent>
    <groupId>io.confluent</groupId>
    <artifactId>kafka-schema-registry-parent</artifactId>
    <version>5.3.0</version>
  </parent>

  <licenses>
    <license>
      <name>Apache License 2.0</name>
      <url>http://www.apache.org/licenses/LICENSE-2.0.html</url>
      <distribution>repo</distribution>
    </license>
  </licenses>

  <artifactId>kafka-avro-serializer</artifactId>
  <packaging>jar</packaging>
  <name>kafka-avro-serializer</name>

  <dependencies>
    <dependency>
      <groupId>org.apache.kafka</groupId>
      <artifactId>kafka_${kafka.scala.version}</artifactId>
      <scope>provided</scope>
    </dependency>
  </dependencies>
</project>
```
1.278 swagger-core 1.5.21
1.278.1 Available under license:
No license file was found, but licenses were detected in source scan.

Bundle-License: http://www.apache.org/licenses/LICENSE-2.0.html

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1001452981_1591661276.22/0/swagger-core-1-5-21-jar/META-INF/MANIFEST.MF

1.279 javax-persistence-api 2.2
1.279.1 Available under license:

Found license 'Eclipse Public License 1.0' in 'Copyright (c) 2008, 2009 Sun Microsystems. All rights reserved. This program and the accompanying materials are made available under the terms of the Eclipse Public License v1.0 and Eclipse Distribution License v. 1.0 The Eclipse Public License is available at http://www.eclipse.org/legal/epl-v10.html and the Eclipse Distribution License is available at'

Found license 'Eclipse Public License 1.0' in 'Copyright (c) 2008 - 2015 Oracle Corporation. All rights reserved. * This program and the accompanying materials are made available under the * terms of the Eclipse Public License v1.0 and Eclipse Distribution License v. 1.0 * The Eclipse Public License is available at http://www.eclipse.org/legal/epl-v10.html * and the Eclipse Distribution License is available at'

Found license 'Eclipse Public License 1.0' in 'Copyright (c) 2008 - 2015 Oracle Corporation. All rights reserved. This program and the accompanying materials are made available under the terms of the Eclipse Public License v1.0 and Eclipse Distribution License v. 1.0 * The Eclipse Public License is available at http://www.eclipse.org/legal/epl-v10.html * and the Eclipse Distribution License is available at'

Found license 'Eclipse Public License 1.0' in 'Copyright (c) 2008 - 2017 Oracle Corporation. All rights reserved. This program and the accompanying materials are made available under the terms of the Eclipse Public License v1.0 and Eclipse Distribution License v. 1.0 * The Eclipse Public License is available at http://www.eclipse.org/legal/epl-v10.html * and the Eclipse Distribution License is available at'

Found license 'Eclipse Public License 1.0' in 'Copyright (c) 2008 - 2017 Oracle Corporation. All rights reserved. * This program and the accompanying materials are made available under the * terms of the Eclipse Public License v1.0 and Eclipse Distribution License v. 1.0 * The Eclipse Public License is available at http://www.eclipse.org/legal/epl-v10.html * and the Eclipse Distribution License is available at'

Found license 'Eclipse Public License 1.0' in 'Copyright (c) 2008 - 2017 Oracle Corporation. All rights reserved. * This program and the accompanying materials are made available under the * terms of the Eclipse Public License v1.0 and Eclipse Distribution License v. 1.0 * The Eclipse Public License is available at http://www.eclipse.org/legal/epl-v10.html * and the Eclipse Distribution License is available at'

Found license 'Eclipse Public License 1.0' in 'Copyright (c) 2011 - 2013 Oracle Corporation. All rights reserved. * This program and the accompanying materials are made available under the * terms of the Eclipse Public License v1.0 and Eclipse Distribution License v. 1.0 * The Eclipse Public License is available at
1.280 slf4j 1.7.32

1.280.1 Available under license:

No license file was found, but licenses were detected in source scan.

/**
 * Copyright (c) 2004-2011 QOS.ch
 * All rights reserved.
 *
 * Permission is hereby granted, free of charge, to any person obtaining
 * a copy of this software and associated documentation files (the
 * "Software"), to deal in the Software without restriction, including
 * without limitation the rights to use, copy, modify, merge, publish,
 * distribute, sublicense, and/or sell copies of the Software, and to
 * permit persons to whom the Software is furnished to do so, subject to
 * the following conditions:
 *
 * The above copyright notice and this permission notice shall be
 * included in all copies or substantial portions of the Software.
 *
 * THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
 * EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF

http://www.eclipse.org/legal/epl-v10.html * and the Eclipse Distribution License is available at'
Found license 'Eclipse Public License 1.0' in 'Copyright (c) 2011 - 2015 Oracle Corporation. All rights reserved. * This program and the accompanying materials are made available under the * terms of the Eclipse Public License v1.0 and Eclipse Distribution License v. 1.0 * The Eclipse Public License is available at http://www.eclipse.org/legal/epl-v10.html * and the Eclipse Distribution License is available at'
Found license 'Eclipse Public License 1.0' in 'Copyright (c) 2008 - 2013 Oracle Corporation. All rights reserved. This program and the accompanying materials are made available under the terms of the Eclipse Public License v1.0 and Eclipse Distribution License v. 1.0 * The Eclipse Public License is available at http://www.eclipse.org/legal/epl-v10.html and the Eclipse Distribution License is available at'
Found license 'Eclipse Public License 1.0' in 'Copyright (c) 2008, 2010 Oracle. All rights reserved. This program and the accompanying materials are made available under the terms of the Eclipse Public License v1.0 and Eclipse Distribution License v. 1.0 which accompanies this distribution. The Eclipse Public License is available'
Found license 'Eclipse Public License 1.0' in 'Copyright (c) 2008 - 2013 Oracle Corporation. All rights reserved. This program and the accompanying materials are made available under the terms of the Eclipse Public License v1.0 and Eclipse Distribution License v. 1.0 * The Eclipse Public License is available at http://www.eclipse.org/legal/epl-v10.html and the Eclipse Distribution License is available at'
Found license 'Eclipse Public License 1.0' in 'Copyright (c) 2017 Oracle Corporation. All rights reserved. * This program and the accompanying materials are made available under the * terms of the Eclipse Public License v1.0 and Eclipse Distribution License v. 1.0 * The Eclipse Public License is available at http://www.eclipse.org/legal/epl-v10.html * and the Eclipse Distribution License is available at'
1.281 netty-codec-smtp 4.1.69.Final

1.281.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*

* Copyright 2016 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1235364851_1638382288.49/0/netty-codec-smtp-4-1-69-final-sources-jar/io/netty/handler/codec/smtp/LastSmtpContent.java
*/
jar/io/netty/handler/codec/smtp/DefaultLastSmtpContent.java

No license file was found, but licenses were detected in source scan.

~ Copyright 2016 The Netty Project
~
~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~
~ https://www.apache.org/licenses/LICENSE-2.0
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.

Found in path(s):

1.282 org.jacoco.agent.rt 0.8.5
1.282.1 Available under license :

Found license 'Eclipse Public License 1.0' in 'This program and the accompanying materials are made available under the terms of the Eclipse Public License 2.0 which is available at'
Found license 'General Public License 2.0' in '<title>JaCoCo - Eclipse Public License - Version 2.0</title> <span class="el_source">Eclipse Public License - Version 2.0</span> <h1>Eclipse Public License - v 2.0</h1> PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION Distributed under this Agreement, and "Secondary License" means either the GNU General Public License, a) Subject to the terms of this Agreement, each Contributor hereby b) Subject to the terms of this Agreement, each Contributor hereby different than this Agreement, provided that such license: trademark, attribution notices, disclaimers of warranty, or
Open Source Used in security_automation identity-guest-cs

limitations'
Found license 'Eclipse Public License 1.0' in 'under the terms and conditions of the Eclipse Public License Version 2.0'
Found license 'Eclipse Public License 2.0' in '<title>JaCoCo - Eclipse Public License - Version 2.0</title> <span class="el_source">Eclipse Public License - Version 2.0</span> <h1>Eclipse Public License - v 2.0</h1> PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION Distributed under this Agreement, and "Secondary License" means either the GNU General Public License, a) Subject to the terms of this Agreement, each Contributor hereby b) Subject to the terms of this Agreement, each Contributor hereby different than this Agreement, provided that such license: trademark, attribution notices, disclaimers of warranty, or limitations'
Found license 'Eclipse Public License 1.0' in 'JaCoCo is a free Java code coverage library distributed under the <a href="doc/license.html">Eclipse Public License</a>. Check'
Found license 'Eclipse Public License 1.0' in '<li>JaCoCo now distributed under the terms and conditions of the Eclipse Public License Version 2.0'
Found license 'Eclipse Public License 1.0' in 'under the terms and conditions of the Eclipse Public License Version 2.0 terms and conditions of the following license: Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies so, subject to the following conditions: The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software. <a href="http://asm.objectweb.org/">ASM 7.2</a> is subject to the terms and conditions of the following license: All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. 3. Neither the name of the copyright holders nor the names of its this software without specific prior written permission. is subject to the terms and conditions of the following license: Apache License http://www.apache.org/licenses/ 2. Grant of Copyright License. Subject to the terms and conditions of 3. Grant of Patent License. Subject to the terms and conditions of attribution notices from the Source form of the Work, include a readable copy of the attribution notices contained do not modify the License. You may add Your own attribution that such additional attribution notices cannot be construed 6. Trademarks. This License does not grant permission to use the trade'
Found license 'Eclipse Public License 1.0' in 'under the terms and conditions of the Eclipse Public License Version 2.0 <a href="http://asm.objectweb.org/">ASM 7.2</a> is subject to the terms and conditions of the following license: All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. 3. Neither the name of the copyright holders nor the names of its this software without specific prior written permission.'
Found license 'Eclipse Public License 1.0' in '* This program and the accompanying materials are made available under * the terms of the Eclipse Public License 2.0 which is available at'
Found license 'Eclipse Public License 1.0' in 'under the terms and conditions of the Eclipse Public License Version 2.0 is subject to the terms and conditions of the following license: Apache License http://www.apache.org/licenses/ 2. Grant of Copyright License. Subject to the terms and conditions of 3. Grant of Patent License. Subject to the terms and conditions of attribution notices from the Source form of the Work, include a readable copy of the attribution notices contained do not modify the License. You may add Your own attribution that such additional attribution notices cannot be construed 6. Trademarks. This License does not grant permission to use the trade'
1.283 janino 3.1.6

1.283.1 Available under license:
No license file was found, but licenses were detected in source scan.

/*
 * Janino - An embedded Java[TM] compiler
 *
 * Copyright (c) 2001-2010 Arno Unkrig. All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without modification, are permitted provided that the
 * following conditions are met:
 *
 * 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the
 *    following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the
 *    following disclaimer in the documentation and/or other materials provided with the distribution.
 * 3. Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote
 *    products derived from this software without specific prior written permission.
 *
 * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND
 * FITNESS FOR A PARTICULAR PURPOSE ARE
 * DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR
 * ANY DIRECT, INDIRECT, INCIDENTAL,
* OPEN SOURCE USED IN SECURITY_AUTOMATION.IDENTITY-GUEST-CS 2712

* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
* PROCUREMENT OF SUBSTITUTE GOODS OR
* SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED
AND ON ANY THEORY OF LIABILITY,
* WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR
OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
* OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

// version.

Found in path(s):
* /opt/cola/permits/1221016761_1635406644.92/0/janino-3-1-6-sources-jar/org/codehaus/janino/Parser.java
No license file was found, but licenses were detected in source scan.

/ *
* Janino - An embedded Java[TM] compiler
*
* Copyright (c) 2001-2018 Arno Unkrig. All rights reserved.
* *
* Redistribution and use in source and binary forms, with or without modification, are permitted provided that the
* following conditions are met:
*
* 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the
following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the
following disclaimer in the documentation and/or other materials provided with the distribution.
* 3. Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote
products derived from this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES,
* INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND
FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR
ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
PROCUREMENT OF SUBSTITUTE GOODS OR
* SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED
AND ON ANY THEORY OF LIABILITY,
* WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR
OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
* OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

Found in path(s):
* /opt/cola/permits/1221016761_1635406644.92/0/janino-3-1-6-sources-jar/org/codehaus/janino/JaninoOption.java
No license file was found, but licenses were detected in source scan.
/*
* Janino - An embedded Java[TM] compiler
*
* Copyright (c) 2001-2020 Arno Unkrig. All rights reserved.
*
* Redistribution and use in source and binary forms, with or without modification, are permitted provided that the
* following conditions are met:
*
* 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the
*    following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the
*    following disclaimer in the documentation and/or other materials provided with the distribution.
* 3. Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote
*    products derived from this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
* ANY EXPRESS OR IMPLIED WARRANTIES,
* INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND
* FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR
* ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
* PROCUREMENT OF SUBSTITUTE GOODS OR
* SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED
* AND ON ANY THEORY OF LIABILITY,
* WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR
* OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
* OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

Found in path(s):
* /opt/cola/permits/1221016761_1635406644.92/0/janino-3-1-6-sources-jar/org/codehaus/janino/StackMap.java
No license file was found, but licenses were detected in source scan.

/*
* Janino - An embedded Java[TM] compiler
*
* Copyright (c) 2016 Arno Unkrig. All rights reserved.
*
* Redistribution and use in source and binary forms, with or without modification, are permitted provided that the
* following conditions are met:
*
* 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the
*    following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the
*    following disclaimer in the documentation and/or other materials provided with the distribution.
* 3. Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote
*    products derived from this software without specific prior written permission.
*/
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
* ANY EXPRESS OR IMPLIED WARRANTIES,
* INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND
FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR
ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
PROCUREMENT OF SUBSTITUTE GOODS OR
* SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED
AND ON ANY THEORY OF LIABILITY,
* WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR
OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
* OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

Found in path(s):
* /opt/cola/permits/1221016761_1635406644.92/0/janino-3-1-6-sources-jar/org/codehaus/janino/TokenType.java
* /opt/cola/permits/1221016761_1635406644.92/0/janino-3-1-6-sources-jar/org/codehaus/janino/TokenStream.java
* /opt/cola/permits/1221016761_1635406644.92/0/janino-3-1-6-sources-jar/org/codehaus/janino/Token.java
* /opt/cola/permits/1221016761_1635406644.92/0/janino-3-1-6-sources-jar/org/codehaus/janino/util/Annotatable.java
No license file was found, but licenses were detected in source scan.

/*
* Janino - An embedded Java[TM] compiler
* *
* Copyright (c) 2018 Arno Unkrig. All rights reserved.
* *
* Redistribution and use in source and binary forms, with or without modification, are permitted provided that the
* following conditions are met:
* *
* 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the
* 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the
* 3. Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote
products derived from this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
* ANY EXPRESS OR IMPLIED WARRANTIES,
* INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND
FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR
ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
PROCUREMENT OF SUBSTITUTE GOODS OR
* SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED
*/
AND ON ANY THEORY OF LIABILITY,
* WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR
OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
* OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
* /

Found in path(s):
* /opt/cola/permits/1221016761_1635406644.92/0/janino-3-1-6-sources-jar/org/codehaus/janino/util/DeepCopier.java
No license file was found, but licenses were detected in source scan.

*/

* Janino - An embedded Java[TM] compiler
*
* Copyright (c) 2013 Arno Unkrig. All rights reserved.
*
* Redistribution and use in source and binary forms, with or without modification, are permitted provided that the
* following conditions are met:
*
* 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the
*    following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the
*    following disclaimer in the documentation and/or other materials provided with the distribution.
* 3. Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote
*    products derived from this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES,
* INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND
FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR
ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
PROCUREMENT OF SUBSTITUTE GOODS OR
* SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED
AND ON ANY THEORY OF LIABILITY,
* WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR
OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
* OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
* /

Found in path(s):
* /opt/cola/permits/1221016761_1635406644.92/0/janino-3-1-6-sources-jar/org/codehaus/janino/package-info.java
* /opt/cola/permits/1221016761_1635406644.92/0/janino-3-1-6-sources-jar/org/codehaus/janino/samples/package-info.java
* /opt/cola/permits/1221016761_1635406644.92/0/janino-3-1-6-sources-jar/org/codehaus/janino/util/package-info.java
* /opt/cola/permits/1221016761_1635406644.92/0/janino-3-1-6-sources-jar/org/codehaus/janino/tools/package-info.java
* /opt/cola/permits/1221016761_1635406644.92/0/janino-3-1-6-sources-jar/org/codehaus/janino/util/package-info.java
* /opt/cola/permits/1221016761_1635406644.92/0/janino-3-1-6-sources-jar/org/codehaus/janino/tools/package-info.java
No license file was found, but licenses were detected in source scan.

/*
 * Janino - An embedded Java[TM] compiler
 *
 * Copyright (c) 2001-2010 Arno Unkrig. All rights reserved.
 * Copyright (c) 2015-2016 TIBCO Software Inc. All rights reserved. // CHECKSTYLE:OFF CHECKSTYLE:ON
 *
 * Redistribution and use in source and binary forms, with or without modification, are permitted provided that the
 * following conditions are met:
 *
 * 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the
 *    following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the
 *    following disclaimer in the documentation and/or other materials provided with the distribution.
 * 3. Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote
 *    products derived from this software without specific prior written permission.
 *
 * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
 * ANY EXPRESS OR IMPLIED WARRANTIES,
 * INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND
 * FITNESS FOR A PARTICULAR PURPOSE ARE
 * DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR
 * ANY DIRECT, INDIRECT, INCIDENTAL,
 * SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
 * PROCUREMENT OF SUBSTITUTE GOODS OR
 * SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED
 * AND ON ANY THEORY OF LIABILITY,
 * WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR
 * OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
 * OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
 */

Found in path(s):
* /opt/cola/permits/1221016761_1635406644.92/0/janino-3-1-6-sources-jar/org/codehaus/janino/CodeContext.java

No license file was found, but licenses were detected in source scan.

/*
 * Janino - An embedded Java[TM] compiler
 *
 * Copyright (c) 2001-2010 Arno Unkrig. All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without modification, are permitted provided that the
 * following conditions are met:
 *
 * 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the
 *    following disclaimer.
 */
* 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the
   following disclaimer in the documentation and/or other materials provided with the distribution.
* 3. Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote
   products derived from this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES,
* INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND
FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR
ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
PROCUREMENT OF SUBSTITUTE GOODS OR
* SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED
AND ON ANY THEORY OF LIABILITY.
* WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR
OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
* OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

Found in path(s):
* /opt/cola/permits/1221016761_1635406644.92/0/janino-3-1-6-sources-jar/org/codehaus/janino/UnicodeUnescapeException.java
* /opt/cola/permits/1221016761_1635406644.92/0/janino-3-1-6-sources-jar/org/codehaus/janino/CompilerFactory.java
* /opt/cola/permits/1221016761_1635406644.92/0/janino-3-1-6-sources-jar/org/codehaus/janino/CachingJavaSourceClassLoader.java
* /opt/cola/permits/1221016761_1635406644.92/0/janino-3-1-6-sources-jar/org/codehaus/janino/Opcode.java
* /opt/cola/permits/1221016761_1635406644.92/0/janino-3-1-6-sources-jar/org/codehaus/janino/ClassBodyEvaluator.java
* /opt/cola/permits/1221016761_1635406644.92/0/janino-3-1-6-sources-jar/org/codehaus/janino/ClassFileIClass.java
* /opt/cola/permits/1221016761_1635406644.92/0/janino-3-1-6-sources-jar/org/codehaus/janino/ClassFileIClassLoader.java
* /opt/cola/permits/1221016761_1635406644.92/0/janino-3-1-6-sources-jar/org/codehaus/janino/Unparser.java
* /opt/cola/permits/1221016761_1635406644.92/0/janino-3-1-6-sources-jar/org/codehaus/janino/samples/DeclarationCounter.java
No license file was found, but licenses were detected in source scan.

/*
 * Janino - An embedded Java[TM] compiler
 *
 * Copyright (c) 2001-2019 Arno Unkrig. All rights reserved.
 * Copyright (c) 2015-2016 TIBCO Software Inc. All rights reserved. // CHECKSTYLE:OFF CHECKSTYLE:ON
 *
 * Redistribution and use in source and binary forms, with or without modification, are permitted provided that the
 * following conditions are met:
 * *
 * 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the
 *    following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the
 *    following disclaimer in the documentation and/or other materials provided with the distribution.
 * 3. Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote
 *    products derived from this software without specific prior written permission.
 */
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, * INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE * DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, * SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR * SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, * WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE * OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

Found in path(s):
* /opt/cola/permits/1221016761_1635406644.92/0/janino-3-1-6-sources-jar/org/codehaus/janino/UnitCompiler.java

1.284 spring-cloud-starter 3.0.4
1.284.1 Available under license :

Apache License
Version 2.0, January 2004
https://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.
"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.
You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.285 asm-tree 7.1

1.285.1 Available under license:

No license file was found, but licenses were detected in source scan.

// All rights reserved.
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions
// are met:
// 1. Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
// 2. Redistributions in binary form must reproduce the above copyright
// notice, this list of conditions and the following disclaimer in the
// documentation and/or other materials provided with the distribution.
// 3. Neither the name of the copyright holders nor the names of its
// this software without specific prior written permission.

Found in path(s):
* /opt/cola/permits/1135892749_1613626907.72/0/asm-tree-7-1-sources-2-jar/org/objectweb/asm/tree/LocalVariableAnnotationNode.java
* /opt/cola/permits/1135892749_1613626907.72/0/asm-tree-7-1-sources-2-jar/org/objectweb/asm/tree/LdcInsnNode.java
* /opt/cola/permits/1135892749_1613626907.72/0/asm-tree-7-1-sources-2-jar/org/objectweb/asm/tree/JumpInsnNode.java
* /opt/cola/permits/1135892749_1613626907.72/0/asm-tree-7-1-sources-2-jar/org/objectweb/asm/tree/AbstractInsnNode.java
* /opt/cola/permits/1135892749_1613626907.72/0/asm-tree-7-1-sources-2-jar/org/objectweb/asm/tree/MethodNode.java
* /opt/cola/permits/1135892749_1613626907.72/0/asm-tree-7-1-sources-2-jar/org/objectweb/asm/tree/ParameterNode.java
* /opt/cola/permits/1135892749_1613626907.72/0/asm-tree-7-1-sources-2-jar/org/objectweb/asm/tree/TypeAnnotationNode.java
* /opt/cola/permits/1135892749_1613626907.72/0/asm-tree-7-1-sources-2-jar/org/objectweb/asm/tree/FieldInsnNode.java
* /opt/cola/permits/1135892749_1613626907.72/0/asm-tree-7-1-sources-2-jar/org/objectweb/asm/tree/AnnotationNode.java
* /opt/cola/permits/1135892749_1613626907.72/0/asm-tree-7-1-sources-2-jar/org/objectweb/asm/tree/InsnList.java
* /opt/cola/permits/1135892749_1613626907.72/0/asm-tree-7-1-sources-2-jar/org/objectweb/asm/tree/InvokeDynamicInsnNode.java
* /opt/cola/permits/1135892749_1613626907.72/0/asm-tree-7-1-sources-2-jar/org/objectweb/asm/tree/ModuleOpenNode.java
* /opt/cola/permits/1135892749_1613626907.72/0/asm-tree-7-1-sources-2-jar/org/objectweb/asm/tree/LocalVariableNode.java
* /opt/cola/permits/1135892749_1613626907.72/0/asm-tree-7-1-sources-2-jar/org/objectweb/asm/tree/FrameNode.java
* /opt/cola/permits/1135892749_1613626907.72/0/asm-tree-7-1-sources-2-jar/org/objectweb/asm/tree/VarInsnNode.java
* /opt/cola/permits/1135892749_1613626907.72/0/asm-tree-7-1-sources-2-jar/org/objectweb/asm/tree/MethodInsnNode.java
* /opt/cola/permits/1135892749_1613626907.72/0/asm-tree-7-1-sources-2-jar/org/objectweb/asm/tree/Util.java
* /opt/cola/permits/1135892749_1613626907.72/0/asm-tree-7-1-sources-2-jar/org/objectweb/asm/tree/MultiANewArrayInsnNode.java
* /opt/cola/permits/1135892749_1613626907.72/0/asm-tree-7-1-sources-2-jar/org/objectweb/asm/tree/LookupSwitchInsnNode.java
* /opt/cola/permits/1135892749_1613626907.72/0/asm-tree-7-1-sources-2-jar/org/objectweb/asm/tree/IincInsnNode.java
* /opt/cola/permits/1135892749_1613626907.72/0/asm-tree-7-1-sources-2-jar/org/objectweb/asm/tree/InsnNode.java
* /opt/cola/permits/1135892749_1613626907.72/0/asm-tree-7-1-sources-2-jar/org/objectweb/asm/tree/InnerClassNode.java
* /opt/cola/permits/1135892749_1613626907.72/0/asm-tree-7-1-sources-2-jar/org/objectweb/asm/tree/ModuleExportNode.java
No license file was found, but licenses were detected in source scan.

2011 INRIA, France Telecom
* All rights reserved.
*  
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in the
* documentation and/or other materials provided with the distribution.
* 3. Neither the name of the copyright holders nor the names of its
* contributors may be used to endorse or promote products derived from
* this software without specific prior written permission.
*  
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
* LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
* CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
* SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
* INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
* CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
1.286 netty-codec-redis 4.1.69.Final

1.286.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2018 The Netty Project
 * 
 * The Netty Project licenses this file to you under the Apache License, version 2.0 (the
 * "License"); you may not use this file except in compliance with the License. You may obtain a
 * copy of the License at:
 * 
 * https://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

No license file was found, but licenses were detected in source scan.

~ Copyright 2016 The Netty Project
~
~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~
~ https://www.apache.org/licenses/LICENSE-2.0
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.
/*
 * Copyright 2016 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License, version 2.0 (the
 * "License"); you may not use this file except in compliance with the License. You may obtain a
 * copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/cola/permits/1235364865_1638382279.47/0/netty-codec-redis-4-1-69-final-sources-jar/io/netty-handler/codec/redis/LastBulkStringRedisContent.java
1.287 scim2-sdk-client 2.1.1

1.287.1 Available under license:

Found license 'GNU Lesser General Public License' in * This program is free software; you can redistribute it and/or modify * it under the terms of the GNU General Public License (GPLv2 only) * or the terms of the GNU Lesser General Public License (LGPLv2.1 only) * as published by the Free Software Foundation. * This program is distributed in the hope that it will be useful, * but WITHOUT ANY WARRANTY; without even the implied warranty of * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the * GNU General Public License for more details. * You should have received a copy of the GNU General Public License * along with this program; if not, see <http://www.gnu.org/licenses/>.'

1.288 spring-data-cassandra 3.1.14

1.288.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2018-2021 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 */
* https://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/ReactiveSelectOperationSupport.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/ReactiveCassandraBatchTemplate.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/cql/CqlOperationsExtensions.kt
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/query/QueryExtensions.kt
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/ExecutableSelectOperation.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/ExecutableInsertOperation.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/mapping/CodecRegistryTupleTypeFactory.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/CassandraTupleValueProvider.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/FluentCassandraOperations.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/query/QueryExtensions.kt
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/ExecutableSelectOperationExtensions.kt
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/ExecutableInsertOperation.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/query/CriteriaExtensions.kt
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/AsyncCassandraOperationsExtensions.kt
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/ExecutableUpdateOperationSupport.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/convert/TupleValueProvider.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/ExecutableSelectOperationExtensions.kt
/*
 * Copyright 2014-2021 the original author or authors.
 * 
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 *      https://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/ReactiveInsertOperation.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/ReactiveInsertOperationExtensions.kt
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/ReactiveDeleteOperation.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/mapping/event/AfterDeleteEvent.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/ExecutableInsertOperationExtensions.kt
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/repository/CountQuery.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/ExecutableUpdateOperation.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/ReactiveFluentCassandraOperations.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/mapping/event/AbstractDeleteEvent.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/mapping/event/AfterConvertEvent.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/mapping/Element.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/ReactiveUpdateOperation.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/ReactiveUpdateOperationExtensions.kt
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/mapping/CassandraPersistentTupleMetadataVerifier.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/mapping/Tuple.java

No license file was found, but licenses were detected in source scan.

*/

/*
 * Copyright 2014-2021 the original author or authors.
 * 
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 *      https://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
Found in path(s):
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/repository/query/PartTreeCassandraQuery.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/repository/cdi/CassandraRepositoryBean.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/repository/cdi/CassandraRepositoryExtension.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2016-2021 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License")
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * *
 *   https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/CassandraOperations.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/AsyncCassandraOperations.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2013-2021 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * *
 *   https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
Found in path(s):
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/mapping/CassandraColumnAnnotationComparator.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/mapping/PrimaryKey.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/mapping/CassandraPersistentEntityMetadataVerifier.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/repository/Query.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/repository/config/EnableCassandraRepositories.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/repository/query/CassandraEntityInformation.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/mapping/CassandraPersistentPropertyComparator.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/mapping/CassandraSimpleTypeHolder.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/mapping/PrimaryKeyColumn.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/mapping/PrimaryKeyClass.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/mapping/PropertyMapping.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/config/CassandraTemplateFactoryBean.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/config/SessionFactoryFactoryBean.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/mapping/MappingCassandraConverter.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/mapping/AbstractCassandraConverter.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/convert/RowReaderPropertyAccessor.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/convert/Mapping.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/mapping/CassandraPersistentPropertyComparator.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/mapping/Indexed.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/mapping/PrimaryKey.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/mapping/CassandraPersistentEntityMetadataVerifier.java

jar/org/springframework/data/cassandra/core/mapping/BasicCassandraPersistentEntityMetadataVerifier.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/mapping/Table.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/convert/CassandraRowValueProvider.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/mapping/Column.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/mapping/BasicCassandraPersistentEntity.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/mapping/VerifierMappingExceptions.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/config/CassandraSessionFactoryBean.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/mapping/CassandraPrimaryKeyColumnAnnotationComparator.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/mapping/BasicCassandraMappingContext.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/repository/support/SimpleCassandraRepository.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/repository/support/MappingCassandraEntityInformation.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/mapping/EntityMapping.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/mapping/CassandraType.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/repository/query/CassandraEntityMetadata.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/repository/support/CassandraRepositoryFactoryBean.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/repository/support/CassandraRepositoryFactory.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2013-2021 the original author or authors.
 */
Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/cql/generator/KeyspaceNameCqlGenerator.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/cql/generator/KeyspaceOption.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/cql/generator/ColumnsOptionCqlGenerator.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/cql/generator/ColumnChangeCqlGenerator.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/cql/generator/DropColumnCqlGenerator.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/cql/generator/TableSpecification.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/cql/generator/DefaultOption.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/cql/generator/TableOptionsSpecification.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/cql/generator/DropIndexCqlGenerator.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/cql/generator/DropTableCqlGenerator.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/core/cql/keyspace/CreateIndexSpecification.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/core/cql/keyspace/KeyNameSpecification.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/core/cql/generator/TableNameCqlGenerator.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/core/cql/keyspace/AddColumnSpecification.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/core/cql/keyspace/ColumnSpecification.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/core/cql/keyspace/TableNameSpecification.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/core/cql/generator/KeyspaceOptionsCqlGenerator.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/core/cql/keyspace/DropKeyspaceSpecification.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/core/cql/keyspace/KeyspaceOptionsSpecification.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/core/cql/keyspace/Option.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/core/cql/keyspace/DropIndexSpecification.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/core/cql/generator/TableOptionsCqlGenerator.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/core/cql/generator/TableCqlGenerator.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/core/cql/generator/TableCqlGenerator.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/core/cql/generator/TableCqlGenerator.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/core/cql/generator/TableCqlGenerator.java
/*
 * Copyright 2019-2021 the original author or authors.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * https://www.apache.org/licenses/LICENSE-2.0
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources.jar/org/springframework/data/cassandra/core/DeleteOptions.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources.jar/org/springframework/data/cassandra/core/mapping/event/AuditingEntityCallback.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources.jar/org/springframework/data/cassandra/core/cql/session/init/CannotReadScriptException.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources.jar/org/springframework/data/cassandra/core/cql/session/init/UncategorizedScriptException.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources.jar/org/springframework/data/cassandra/core/cql/session/init/SessionFactoryInitializer.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources.jar/org/springframework/data/cassandra/core/cql/session/init/CompositeKeyspacePopulator.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources.jar/org/springframework/data/cassandra/config/CassandraAuditingRegistrar.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources.jar/org/springframework/data/cassandra/core/cql/session/init/SessionFactoryInitializer.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources.jar/org/springframework/data/cassandra/core/cql/session/init/CassandraFutureAdapter.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources.jar/org/springframework/data/cassandra/core/cql/session/init/CompositeKeyspacePopulator.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources.jar/org/springframework/data/cassandra/core/cql/session/init/ScriptUtils.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources.jar/org/springframework/data/cassandra/core/cql/keyspace/AlterTableSpecification.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources.jar/org/springframework/data/cassandra/core/cql/keyspace/KeyspaceActionSpecification.java

No license file was found, but licenses were detected in source scan.
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2016-2021 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/repository/query/ReactivePartTreeCassandraQuery.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/cql/generator/UserTypeNameCqlGenerator.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/cql/keyspace/AlterUserTypeSpecification.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/repository/support/ReactiveCassandraRepositoryFactory.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/mapping/SimpleUserTypeResolver.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/mapping/CompositeCassandraPersistentEntityMetadataVerifier.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/cql/CqlProvider.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/cql/userdefinedtype/UserDefinedType.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/cql/ArgumentPreparedStatementBinder.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/ReactiveCassandraOperations.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/cql/keyspace/UserTypeNameSpecification.java

Open Source Used In security Automation Identity-guest-cs 2739
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources.jar/org/springframework/repository/query/CassandraParameterAccessor.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources.jar/org/springframework/repository/query/CassandraParametersParameterAccessor.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources.jar/org/springframework/core/convert/CassandraThreeTenBackPortConverters.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources.jar/org/springframework/core/cql/keyspace/CreateUserTypeSpecification.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources.jar/org/springframework/core/cql/keyspace/RenameColumnSpecification.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources.jar/org/springframework/core/cql/generator/DropUserTypeCqlGenerator.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources.jar/org/springframework/core/cql/generator/InsertUserTypeCqlGenerator.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources.jar/org/springframework/core/cql/generator/UpdateUserTypeCqlGenerator.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources.jar/org/springframework/core/convert/RowReader.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources.jar/org/springframework/core/ReactiveCassandraTemplate.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources.jar/org/springframework/core/mapping/PrimaryKeyClassEntityMetadataVerifier.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources.jar/org/springframework/core/ReactiveCassandraRepositoryFactoryBean.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources.jar/org/springframework/core/ReactiveCassandraRepositoryConfigurationExtension.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources.jar/org/springframework/core/ReactiveCassandraRepositoryConfiguration.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources.jar/org/springframework/core/ReactiveCassandraEntityMetadata.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources.jar/org/springframework/core/cql/keyspace/CreateColumnSpecification.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources.jar/org/springframework/core/cql/keyspace/RenameColumnSpecification.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources.jar/org/springframework/core/cql/keyspace/CreateUserTypeSpecification.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2021 the original author or authors.
 */
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-
jar/org/springframework/data/cassandra/core/mapping/CachingCassandraPersistentProperty.java
No license file was found, but licenses were detected in source scan.

*/

* Copyright 2020-2021 the original author or authors.
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-
jar/org/springframework/data/cassandra/core/mapping/Frozen.java
No license file was found, but licenses were detected in source scan.

*/

* Copyright 2018-2021 the original author or authors.
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-
jar/org/springframework/data/cassandra/core/mapping/event/AbstractCassandraEventListener.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/mapping/event/CassandraMappingEvent.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2017-2021 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/cql/keyspace/KeyspaceAttributes.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/repository/MapIdCassandraRepository.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/cql/converter/ResultSetToByteBufferConverter.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/cql/converter/ResultSetToIntegerConverter.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/cql/converter/ResultSetToArrayConverter.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/repository/query/QueryStatementCreator.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/cql/converter/ResultSetToBooleanConverter.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/cql/generator/OptionRenderUtil.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/cql/CqlConstantType.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/mapping/IdInterfaceExceptions.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/config/CqlNamespaceHandler.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/mapping/NamingStrategy.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/config/KeySpaceActionSpecificationFactory.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2020-2021 the original author or authors.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/convert/CassandraColumnType.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/repository/query/BindingContext.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/mapping/SnakeCaseNamingStrategy.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/convert/UserTypeUtil.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/cql/DataClassRowMapper.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/convert/ColumnType.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/convert/DefaultColumnType.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/convert/DefaultColumnTypeResolver.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/cql/DataClassRowMapperExtensions.kt
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/cql/ExecutionProfileResolver.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/config/EnableReactiveCassandraAuditing.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/convert/ColumnTypeResolver.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/mapping/TransformingNamingStrategy.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/mapping/Embedded.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/cql/QueryExtractorDelegate.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/repository/query/DefaultSpELExpressionEvaluator.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/convert/SchemaFactory.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/convert/DefaultCassandraColumnType.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/core/convert/FrozenIndicator.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/config/AbstractCassandraConfiguration.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/config/AbstractCassandraConfiguration.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/config/PersistentEntitiesFactoryBean.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2013-2021 the original author or authors.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * https://www.apache.org/licenses/LICENSE-2.0
 */

* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/config/AbstractCassandraConfiguration.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-jar/org/springframework/data/cassandra/CassandraKeyspaceExistsException.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-
Open Source Used In security_automation identity-guest-cs 2751

* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-
jar/org/springframework/data/cassandra/core/cql/RowCallbackHandler.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-
jar/org/springframework/data/cassandra/CassandraSchemaElementExistsException.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-
jar/org/springframework/data/cassandra/config/CassandraMappingConverterParser.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-
jar/org/springframework/data/cassandra/core/CassandraAdminTemplate.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-
jar/org/springframework/data/cassandra/core/cql/CachedPreparedStatementCreator.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-
jar/org/springframework/data/cassandra/config/CassandraCqlTemplateFactoryBean.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-
jar/org/springframework/data/cassandra/core/CassandraAdminOperations.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-
jar/org/springframework/data/cassandra/core/cql/keyspace/TableOption.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-
jar/org/springframework/data/cassandra/core/cql/CassandraAccessor.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-
jar/org/springframework/data/cassandra/config/SessionBuilderConfigurer.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-
jar/org/springframework/data/cassandra/core/cql/CassandraTableExistsException.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-
jar/org/springframework/data/cassandra/config/ParsingUtils.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-
jar/org/springframework/data/cassandra/core/convert/BasicCassandraRowValueProvider.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-
jar/org/springframework/data/cassandra/core/cql/WriteOptions.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-
jar/org/springframework/data/cassandra/core/cql/Ordering.java
jar/org/springframework/data/cassandra/CassandraTruncateException.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-
jar/org/springframework/data/cassandra/core/convert/RowValueProvider.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-
jar/org/springframework/data/cassandra/CassandraInvalidQueryException.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-
jar/org/springframework/data/cassandra/config/KeyspaceAction.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-
jar/org/springframework/data/cassandra/config/CassandraTemplateParser.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-
jar/org/springframework/data/cassandra/CassandraUncategorizedException.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-
jar/org/springframework/data/cassandra/core/cql/CassandraExceptionTranslator.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-
jar/org/springframework/data/cassandra/core/cql/AsyncResultSetExtractor.java

No license file was found, but licenses were detected in source scan.

/ *
* Copyright 2010-2021 the original author or authors.
* *
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* *
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-
jar/org/springframework/data/cassandra/repository/query/CassandraQueryCreator.java
* /opt/cola/permits/1221016641_1635406653.11/0/spring-data-cassandra-3-1-14-sources-
jar/org/springframework/data/cassandra/repository/query/AbstractCassandraQuery.java

1.289 jettison 1.1
1.289.1 Available under license :

Apache License
Version 2.0, January 2004
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and
(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

Copyright 2006 Envoi Solutions LLC

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
1.290 moshi 1.8.0

1.290.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2015 Square, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
  */opt/cola/permits/1199756599_1630579387.69/0/moshi-1-8-0-sources-3-jar/com/squareup/moshi/ToJson.java
  */opt/cola/permits/1199756599_1630579387.69/0/moshi-1-8-0-sources-3-jar/com/squareup/moshi/FromJson.java
  */opt/cola/permits/1199756599_1630579387.69/0/moshi-1-8-0-sources-3-jar/com/squareup/moshi/JsonDataException.java
  */opt/cola/permits/1199756599_1630579387.69/0/moshi-1-8-0-sources-3-jar/com/squareup/moshi/JsonQualifier.java
  */opt/cola/permits/1199756599_1630579387.69/0/moshi-1-8-0-sources-3-jar/com/squareup/moshi/AdapterMethodsFactory.java
  */opt/cola/permits/1199756599_1630579387.69/0/moshi-1-8-0-sources-3-jar/com/squareup/moshi/MapJsonAdapter.java
  */opt/cola/permits/1199756599_1630579387.69/0/moshi-1-8-0-sources-3-jar/com/squareup/moshi/ClassJsonAdapter.java
  */opt/cola/permits/1199756599_1630579387.69/0/moshi-1-8-0-sources-3-jar/com/squareup/moshi/Json.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2010 The Android Open Source Project
 * Copyright (C) 2012 Google Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 */
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1199756599_1630579387.69/0/moshi-1-8-0-sources-3-
  jar/com/squareup/moshi/LinkedHashTreeMap.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2016 Square, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1199756599_1630579387.69/0/moshi-1-8-0-sources-3-
  jar/com/squareup/moshi/JsonEncodingException.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2011 Google Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
Found in path(s):
* /opt/cola/permits/1199756599_1630579387.69/0/moshi-1-8-0-sources-3-jar/com/squareup/moshi/ClassFactory.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2010 Google Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1199756599_1630579387.69/0/moshi-1-8-0-sources-3-jar/com/squareup/moshi/JsonUtf8Reader.java
* /opt/cola/permits/1199756599_1630579387.69/0/moshi-1-8-0-sources-3-jar/com/squareup/moshi/JsonScope.java
* /opt/cola/permits/1199756599_1630579387.69/0/moshi-1-8-0-sources-3-jar/com/squareup/moshi/JsonUtf8Writer.java
* /opt/cola/permits/1199756599_1630579387.69/0/moshi-1-8-0-sources-3-jar/com/squareup/moshi/JsonReader.java
* /opt/cola/permits/1199756599_1630579387.69/0/moshi-1-8-0-sources-3-jar/com/squareup/moshi/JsonWriter.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2008 Google Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
/*
 * Copyright (C) 2017 Square, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *      http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1199756599_1630579387.69/0/moshi-1-8-0-sources-3-jar/com/squareup/moshi/Types.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2018 Square, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *      http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1199756599_1630579387.69/0/moshi-1-8-0-sources-3-jar/com/squareup/moshi/JsonValueReader.java
* /opt/cola/permits/1199756599_1630579387.69/0/moshi-1-8-0-sources-3-jar/com/squareup/moshi/JsonValueWriter.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2018 Square, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *      http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1199756599_1630579387.69/0/moshi-1-8-0-sources-3-jar/com/squareup/moshi/JsonClass.java
No license file was found, but licenses were detected in source scan.

/
Copyright (C) 2014 Square, Inc.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Found in path(s):

* /opt/cola/permits/1199756599_1630579387.69/0/moshi-1-8-0-sources-3-jar/com/squareup/moshi/CollectionJsonAdapter.java
* /opt/cola/permits/1199756599_1630579387.69/0/moshi-1-8-0-sources-3-jar/com/squareup/moshi/internal/Util.java
* /opt/cola/permits/1199756599_1630579387.69/0/moshi-1-8-0-sources-3-jar/com/squareup/moshi/JsonAdapter.java
* /opt/cola/permits/1199756599_1630579387.69/0/moshi-1-8-0-sources-3-jar/com/squareup/moshi/Moshi.java
* /opt/cola/permits/1199756599_1630579387.69/0/moshi-1-8-0-sources-3-jar/com/squareup/moshi/StandardJsonAdapters.java
* /opt/cola/permits/1199756599_1630579387.69/0/moshi-1-8-0-sources-3-jar/com/squareup/moshi/ArrayJsonAdapter.java

1.291 mixpanel-java 1.5.0
1.291.1 Available under license:

Found license 'GNU Lesser General Public License' in '/This module is multi-licensed and may be used under the terms //EPL, Eclipse Public License, V1.0 or later, http://www.eclipse.org/legal //LGPL, GNU Lesser General Public License, V2.1 or later, http://www.gnu.org/licenses/lgpl.html //GPL, GNU General Public License, V2 or later, http://www.gnu.org/licenses/gpl.html //AL, Apache License, V2.0 or later, http://www.apache.org/licenses //BSD, BSD License, http://www.opensource.org/licenses/bsd-license.php'

Found license 'Eclipse Public License 1.0' in '/This module is multi-licensed and may be used under the terms //EPL, Eclipse Public License, V1.0 or later, http://www.eclipse.org/legal //LGPL, GNU Lesser General Public License, V2.1 or later, http://www.gnu.org/licenses/lgpl.html //GPL, GNU General Public License, V2 or later, http://www.gnu.org/licenses/gpl.html //AL, Apache License, V2.0 or later, http://www.apache.org/licenses //BSD, BSD License, http://www.opensource.org/licenses/bsd-license.php'

1.292 stax 4.0.0
1.292.1 Available under license:

GNU GENERAL PUBLIC LICENSE

Version 3, 29 June 2007

Copyright © 2007 Free Software Foundation, Inc. <https://fsf.org/>

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program--to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it:

responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict
development and use of software on general-purpose computers, but in those that do, we wish to avoid the special
danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures
that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

“This License” refers to version 3 of the GNU General Public License.

“Copyright” also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

“The Program” refers to any copyrightable work licensed under this License. Each licensee is addressed as “you”.
“Licensees” and “recipients” may be individuals or organizations.

To “modify” a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission,
other than the making of an exact copy. The resulting work is called a “modified version” of the earlier work or a
work “based on” the earlier work.

A “covered work” means either the unmodified Program or a work based on the Program.

To “propagate” a work means to do anything with it that, without permission, would make you directly or
secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a
private copy. Propagation includes copying, distribution (with or without modification), making available to the
public, and in some countries other activities as well.

To “convey” a work means any kind of propagation that enables other parties to make or receive copies. Mere
interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays “Appropriate Legal Notices” to the extent that it includes a convenient and
prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no
warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under
this License, and how to view a copy of this License. If the interface presents a list of user commands or options,
such as a menu, a prominent item in the list meets this criterion.


The “source code” for a work means the preferred form of the work for making modifications to it. “Object code”
means any non-source form of a work.

A “Standard Interface” means an interface that either is an official standard defined by a recognized standards body,
or, in the case of interfaces specified for a particular programming language, one that is widely used among
developers working in that language.

The “System Libraries” of an executable work include anything, other than the work as a whole, that (a) is included
in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves
only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A “Major Component”, in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The “Corresponding Source” for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.

You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

a) The work must carry prominent notices stating that you modified it, and giving a relevant date.
b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to “keep intact all notices”.
c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.
d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an “aggregate” if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.
b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.
c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.
d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.
e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A “User Product” is either (1) a “consumer product”, which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, “normally used” refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

“Installation Information” for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

“Additional permissions” are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though
they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or
b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or
c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or
d) Limiting the use for publicity purposes of names of licensors or authors of the material; or
e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or
f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered “further restrictions” within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies
you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.


You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An “entity transaction” is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A “contributor” is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's “contributor version”.

A contributor's “essential patent claims” are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, “control” includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.
In the following three paragraphs, a “patent license” is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To “grant” such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. “Knowingly relying” means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is “discriminatory” if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.


If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through
a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License “or any later version” applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.
END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to
achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to
most effectively state the exclusion of warranty; and each file should have at least the “copyright” line and a pointer
to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year>  <name of author>

This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation, either version 3 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program.  If not, see <https://www.gnu.org/licenses/>.

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short notice like this when it starts in an interactive mode:

<program>  Copyright (C) <year>  <name of author>
This program comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License.
Of course, your program's commands might be different; for a GUI interface, you would use an “about box”.

You should also get your employer (if you work as a programmer) or school, if any, to sign a “copyright disclaimer”
for the program, if necessary. For more information on this, and how to apply and follow the GNU GPL, see
<https://www.gnu.org/licenses/>.

The GNU General Public License does not permit incorporating your program into proprietary programs. If your
program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the
library. If this is what you want to do, use the GNU Lesser General Public License instead of this License. But first,
/**
 * Copyright (C) cedarsoft GmbH.
 *
 * Licensed under the GNU General Public License version 3 (the "License")
 * with Classpath Exception; you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at
 *
 * http://www.cedarsoft.org/gpl3ce
 * (GPL 3 with Classpath Exception)
 *
 * This code is free software; you can redistribute it and/or modify it
 * under the terms of the GNU General Public License version 3 only, as
 * published by the Free Software Foundation. cedarsoft GmbH designates this
 * particular file as subject to the "Classpath" exception as provided
 * by cedarsoft GmbH in the LICENSE file that accompanied this code.
 *
 * This code is distributed in the hope that it will be useful, but WITHOUT
 * ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or
 * FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License
 * version 3 for more details (a copy is included in the LICENSE file that
 * accompanied this code).
 *
 * You should have received a copy of the GNU General Public License version
 * 3 along with this work; if not, write to the Free Software Foundation,
 * Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA.
 *
 * Please contact cedarsoft GmbH, 72810 Gomaringen, Germany,
 * or visit www.cedarsoft.com if you need additional information or
 * have any questions.
 */

1.293 barchart-udt-bundle 2.3.0
1.293.1 Available under license :

====
Copyright (C) 2009-2013 Barchart, Inc. <http://www.barchart.com/>

All rights reserved. Licensed under the OSI BSD License.

http://www.opensource.org/licenses/bsd-license.php
====

#
#
#{projectBuildStamp}
#

---
1.294 jersey-spring 1.19.1
1.294.1 Available under license:

Found license 'General Public License 2.0' in 

* Copyright (c) 2010-2011 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You * may not use this file except in compliance with the License. You can * Oracle designates this particular file as subject to the "Classpath" * exception as provided by Oracle in the GPL Version 2 section of the License

Found license 'General Public License 2.0' in 

* Copyright (c) 2010-2013 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You * may not use this file except in compliance with the License. You can * Oracle designates this particular file as subject to the "Classpath" * exception as provided by Oracle in the GPL Version 2 section of the License

1.295 swagger-jaxrs 1.5.21

1.296 jboss-logging 3.4.2.Final
1.296.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.
"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.297 spring-web-mvc 5.3.12

1.297.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2021 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * *
 * https://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/mvc/method/annotation/RequestPartMethodArgumentResolver.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/mvc/method/annotation/RequestBodyAdvice.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/config/annotation/PathMatchConfigurer.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/resource/ResourceUrlParser.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/mvc/LastModified.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/mvc/HandlerMappingIntrospector.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/handler/HandlerMappingIntrospector.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/mvc/method/annotation/RequestBodyAdviceAdapter.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/mvc/SimpleControllerHandlerAdapter.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/mvc/method/annotation/MvcUriComponentsBuilder.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/function/DefaultEntityResponseBuilder.java
jar/org/springframework/web/servlet/config/annotation/WebMvcConfigurer.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/mvc/Controller.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/mvc/Controller.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/mvc/method/RequestMappingInfoHandlerMapping.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/view/DefaultRequestToViewNameTranslator.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/function/ErrorHandlerServerResponse.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/mvc/method/annotation/SseEmitter.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/handler/MatchableHandlerMapping.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/handler/AbstractHandlerMethodMapping.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/config/annotation/WebMvcConfigurationSupport.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/resource/PathResourceResolver.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/mvc/method/annotation/ServletInvocableHandlerMethod.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/DispatcherServlet.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/function/RequestPredicate.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/function/PathResourceLookupFunction.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/function/support/RouterFunctionMapping.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/config/annotation/CorsRegistration.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/mvc/method/annotation/AbstractMessageConverterMethodArgumentResolver.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/mvc/condition/ParamsRequestCondition.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/mvc/method/annotation/AbstractMessageConverterMethodProcessor.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/mvc/method/condition/HeadersRequestCondition.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/handler/AbstractUrlHandlerMapping.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/i18n/AcceptHeaderLocaleResolver.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2002-2015 the original author or authors.
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

/*
* Copyright 2002-2019 the original author or authors.
*/

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and

/*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*/

https://www.apache.org/licenses/LICENSE-2.0

/*
* Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
*/

| Open Source Used In security_automation identity-guest-cs | 2781 |
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/handler/BeanNameUrlHandlerMapping.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/mvc/ServletWrappingController.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/mvc/method/annotation/StreamingResponseBodyReturnValueHandler.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/mvc/method/annotation/ViewNameMethodReturnValueHandler.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/function/ServerResponseExtensions.kt
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/resource/EncodedResourceExtensions.kt
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/resource/CachingResourceResolver.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/view/script/ScriptTemplateConfig.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/function/ToStringVisitor.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/mvc/method/annotation/ViewMethodReturnValueHandler.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/function/HandlerFunction.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/config/annotation/DelegatingWebMvcConfiguration.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/resource/CachingResourceResolver.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/view/script/ScriptTemplateView.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/function/ServerResponse.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/handler/HandlerExceptionResolverComposite.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/function/RouterFunctionDsL.kt
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/mvc/method/annotation/DeferredResultMethodReturnValueHandler.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/function/EntityResponse.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/mvc/method/annotation/SessionAttributeMethodArgumentResolver.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/function/DefaultRenderingResponseBuilder.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/function/RenderingResponse.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/function/HandlerFilterFunction.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2007 the original author or authors.
 * 
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/view/script/ScriptTemplateConfigurer.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/view/AbstractCachingViewResolver.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/function/ServerRequestExtensions.kt
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/mvc/method/RequestMappingInfoHandlerMethodMappingNamingStrategy.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/resource/WebJarsResourceResolver.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/support/ RequestContext.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/resource/CssLinkResourceTransformer.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/config/annotation/DefaultServletHandlerConfigurer.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/mvc/condition/AbstractRequestCondition.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/function/ResourceHandlerFunction.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/resource/CachingResourceTransformer.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/config/annotation/AsyncSupportConfigurer.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/function/RouterFunction.java

Open Source Used In security_automation identity-guest-cs 2783
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2018 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *   https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

/**
 **
 * A contract for inspecting and potentially modifying request data values such
 * as URL query parameters or form field values before they are rendered by a
 * view or before a redirect.
 *
 * Implementations may use this contract for example as part of a solution
 * to provide data integrity, confidentiality, protection against cross-site
 * request forgery (CSRF), and others or for other tasks such as automatically
 * adding a hidden field to all forms and URLs.
 *
 * View technologies that support this contract can obtain an instance to
 * delegate to via RequestContext#getRequestDataValueProcessor()
 *
 * @author Rossen Stoyanchev
 * @since 3.1
 */

Found in path(s):
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/support/RequestDataValueProcessor.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2016 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *   https://www.apache.org/licenses/LICENSE-2.0
 */
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/resource/ResourceTransformer.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/config/MvcNamespaceHandler.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/config/InterceptorsBeanDefinitionParser.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/config/DefaultServletHandlerBeanDefinitionParser.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/config/annotation/WebMvcConfigurerComposite.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/mvc/method/annotation/StreamingResponseBody.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/config/annotation/WebMvcConfigurerAdapter.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/view/AbstractTemplateView.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/view/tiles3/SpringLocaleResolver.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/mvc/condition/RequestCondition.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2014 the original author or authors.
 * 
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 *     https://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
*/
Found in path(s):
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/FlashMapManager.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/config/annotation/UrlBasedViewResolverRegistration.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/mvc/method/annotation/AbstractMappingJacksonResponseBodyAdvice.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/tags/form/PasswordInputTag.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/support/SessionFlashMapManager.java
No license file was found, but licenses were detected in source scan.

/*/  
* Copyright 2002-2013 the original author or authors.
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
*     https://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/i18n/AbstractLocaleResolver.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-jar/org/springframework/web/servlet/tags/form/HiddenInputTag.java
No license file was found, but licenses were detected in source scan.

/*/  
* Copyright 2002-2010 the original author or authors.
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
*     https://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-
  jar/org/springframework/web/servlet/handler/DispatcherServletWebRequest.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2002-2017 the original author or authors.
* Licensed under the Apache License, Version 2.0 (the "License"
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* https://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-
  jar/org/springframework/web/servlet/mvc/method/annotation/HttpHeadersReturnValueHandler.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-
  jar/org/springframework/web/servlet/resource/AbstractResourceResolver.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-
  jar/org/springframework/web/servlet/view/tiles3/SpringWildcardServletTilesApplicationContext.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-
  jar/org/springframework/web/servlet/ThemeResolver.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-
  jar/org/springframework/web/servlet/mvc/support/RedirectAttributesModelMap.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-
  jar/org/springframework/web/servlet/handler/WebRequestHandlerInterceptorAdapter.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-
  jar/org/springframework/web/servlet/view/tiles3/TilesView.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-
  jar/org/springframework/web/servlet/view/feed/AbstractRssFeedView.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-
  jar/org/springframework/web/servlet/view/JspView.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-
  jar/org/springframework/web/servlet/AsyncHandlerInterceptor.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-
  jar/org/springframework/web/servlet/resource/HttpResource.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2020 the original author or authors.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * https://www.apache.org/licenses/LICENSE-2.0
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-
  jar/org/springframework/web/servlet/mvc/method/annotation/PathVariableMapMethodArgumentResolver.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources.jar/jar/org/springframework/web/servlet/config/annotation/InterceptorRegistration.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources.jar/jar/org/springframework/web/servlet/function/AbstractServerResponse.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources.jar/jar/org/springframework/web/servlet/tags/form/ErrorsTag.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources.jar/jar/org/springframework/web/servlet/function/AbstractServerResponse.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources.jar/jar/org/springframework/web/servlet/annotation/PrincipalMethodArgumentResolver.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources.jar/jar/org/springframework/web/servlet/config/ResourcesBeanDefinitionParser.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources.jar/jar/org/springframework/web/servlet/HandlerExecutionChain.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources.jar/jar/org/springframework/web/servlet/config/AnnotationDrivenBeanDefinitionParser.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2018 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *    https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-
jar/org/springframework/web/servlet/handler/SimpleMappingExceptionResolver.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-
jar/org/springframework/web/servlet/tags/EvalTag.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-
jar/org/springframework/web/servlet/tags/form/AbstractHtmlElementBodyTag.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-
jar/org/springframework/web/servlet/tags/TransformTag.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-
jar/org/springframework/web/servlet/resource/ResourceTransformerChain.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-
jar/org/springframework/web/servlet/theme/CookieThemeResolver.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-
jar/org/springframework/web/servlet/tags/LabelTag.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-
jar/org/springframework/web/servlet/ModelAndView.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-
jar/org/springframework/web/servlet/view/ContentNegotiatingViewResolver.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-
jar/org/springframework/web/servlet/tags/HtmlEscapeTag.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-
jar/org/springframework/web/servlet/HttpServletBean.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-
jar/org/springframework/web/servlet/view/groovy/GroovyMarkupConfigurer.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources-
jar/org/springframework/web/servlet/view/document/AbstractPdfView.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources.jar/org/springframework/web/servlet/mvc/method/annotation/ResponseBodyMethodProcessor.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources.jar/org/springframework/web/servlet/tags/form/OptionTag.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources.jar/org/springframework/web/servlet/tags/ParamTag.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources.jar/org/springframework/web/servlet/tags/NestedPathTag.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources.jar/org/springframework/web/servlet/support/AbstractDispatcherServletInitializer.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources.jar/org/springframework/web/servlet/tags/form/CheckboxesTag.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources.jar/org/springframework/web/servlet/tags/ButtonTag.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources.jar/org/springframework/web/servlet/mvc/method/annotation/RequestResponseBodyMethodProcessor.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources.jar/org/springframework/web/servlet/tags/form/CheckboxTag.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources.jar/org/springframework/web/servlet/mvc/method/annotation/MatrixVariableMapMethodArgumentResolver.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources.jar/org/springframework/web/servlet/tags/RequestContextAwareTag.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources.jar/org/springframework/web/servlet/resource/ContentVersionStrategy.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources.jar/org/springframework/web/servlet/mvc/condition/AbstractNameValueExpression.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources.jar/org/springframework/web/servlet/tags/form/RadioButtonTag.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources.jar/org/springframework/web/servlet/tags/form/ButtonTag.java
* /opt/cola/permits/1220178444_1635319414.1/0/spring-webmvc-5-3-12-sources.jar/org/springframework/web/servlet/mvc/condition/NameValueExpression.java
/*
   Copyright 2002-2017 the original author or authors.
*/

/* Licensed under the Apache License, Version 2.0 (the "License");
   you may not use this file except in compliance with the License.
   You may obtain a copy of the License at
*/

/* https://www.apache.org/licenses/LICENSE-2.0 */
XSLT-driven View that allows for response context to be rendered as the result of an XSLT transformation.

The XSLT Source object is supplied as a parameter in the model and then detected during response rendering. Users can either specify a specific entry in the model via the sourceKey property or have Spring locate the Source object. This class also provides basic conversion of objects into Source implementations. See for more details.

All model parameters are passed to the XSLT Transformer as parameters. In addition the user can configure output properties to be passed to the Transformer.

@since 2.0

Found in path(s):

No license file was found, but licenses were detected in source scan.
1.298 java-driver-metrics-micrometer 4.10.0

1.298.1 Available under license:

No license file was found, but licenses were detected in source scan.

Copyright DataStax, Inc.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Found in path(s):
* /opt/ cola/permits/ 1147203691_1616929112.9/0/java-driver-metricsmicrometer-4-10-0-sources-jar/META-INF/maven/com.datastax.oss/java-driver-metrics-micrometer/pom.xml
No license file was found, but licenses were detected in source scan.
/*
 * Copyright DataStax, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1147203691_1616929112.9/0/java-driver-metrics-micrometer-4-10-0-sources-jar/com/datastax/oss/driver/internal/metrics/micrometer/MicrometerSessionMetricUpdater.java
* /opt/cola/permits/1147203691_1616929112.9/0/java-driver-metrics-micrometer-4-10-0-sources-jar/com/datastax/oss/driver/internal/metrics/micrometer/MicrometerMetricsFactory.java
* /opt/cola/permits/1147203691_1616929112.9/0/java-driver-metrics-micrometer-4-10-0-sources-jar/com/datastax/oss/driver/internal/metrics/micrometer/MicrometerNodeMetricUpdater.java
* /opt/cola/permits/1147203691_1616929112.9/0/java-driver-metrics-micrometer-4-10-0-sources-jar/com/datastax/oss/driver/internal/metrics/micrometer/MicrometerMetricUpdater.java

# 1.299 jackson-jaxrs 2.13.0

## 1.299.1 Available under license:

This copy of Jackson JSON processor databind module is licensed under the Apache (Software) License, version 2.0 ("the License").

See the License for details about distribution rights, and the specific rights regarding derivate works.

You may obtain a copy of the License at:

[http://www.apache.org/licenses/LICENSE-2.0](http://www.apache.org/licenses/LICENSE-2.0)

# Jackson JSON processor

Jackson is a high-performance, Free/Open Source JSON processing library. It was originally written by Tatu Saloranta (tatu.saloranta@iki.fi), and has been in development since 2007. It is currently developed by a community of developers, as well as supported commercially by FasterXML.com.

## Licensing
Jackson core and extension components may be licensed under different licenses. To find the details that apply to this artifact see the accompanying LICENSE file. For more information, including possible other licensing options, contact FasterXML.com (http://fasterxml.com).

## Credits

A list of contributors may be found from CREDITS file, which is included in some artifacts (usually source distributions); but is always available from the source code management (SCM) system project uses.

### 1.300 netty-codec-haproxy 4.1.69.Final

#### 1.300.1 Available under license:

No license file was found, but licenses were detected in source scan.

~ Copyright 2012 The Netty Project
~
~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~
~ https://www.apache.org/licenses/LICENSE-2.0
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.

Found in path(s):
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014 The Netty Project
 * 
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * 
 * https://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
1.301 jaxen 1.2.0

1.301.1 Available under license:

/*
 StatelessWidget
*$Id$

Copyright 2003-2006 The Werken Company. All Rights Reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright
  notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

* Neither the name of the Jaxen Project nor the names of its
  contributors may be used to endorse or promote products derived
  from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS
IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED
TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A
PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER
OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/

1.302 metrics-influxdb 1.1.0

1.302.1 Available under license:
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2014 the original author or authors.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1256060442_1642635485.54/0/pippo-metrics-influxdb-1-1-0-sources-jar/ro/pippo/influxdb/Reporter.java

1.303 asm-commons 7.1

1.303.1 Available under license:
No license file was found, but licenses were detected in source scan.

// All rights reserved.
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions
// are met:
// 1. Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
// 2. Redistributions in binary form must reproduce the above copyright
// notice, this list of conditions and the following disclaimer in the
// documentation and/or other materials provided with the distribution.
// 3. Neither the name of the copyright holders nor the names of its
// this software without specific prior written permission.
/**
 * Constructs a new {@link LocalVariablesSorter}. <i>Subclasses must not use this constructor</i>.
 * Instead, they must use the {@link #LocalVariablesSorter(int, int, String, MethodVisitor)}
 * version.
 *
 * @param access access flags of the adapted method.
 * @param descriptor the method's descriptor (see @link Type{}).
 * @param methodVisitor the method visitor to which this adapter delegates calls.
 * @throws IllegalStateException if a subclass calls this constructor.
 */

Found in path(s):
* /opt/cola/permits/1135892456_1613626901.70/asm-commons-7-1-sources-2-
jar/org/objectweb/asm/commons/LocalVariablesSorter.java
No license file was found, but licenses were detected in source scan.

// All rights reserved.
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions
// are met:
// 1. Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
// 2. Redistributions in binary form must reproduce the above copyright
// notice, this list of conditions and the following disclaimer in the
// documentation and/or other materials provided with the distribution.
// 3. Neither the name of the copyright holders nor the names of its
// this software without specific prior written permission.
/**
 * Constructs a new {@link GeneratorAdapter}. <i>Subclasses must not use this constructor</i>.
 * Instead, they must use the {@link #GeneratorAdapter(int, MethodVisitor, int, String, String)}
 * version.
 *
 * @param methodVisitor the method visitor to which this adapter delegates calls.
 * @param access the method's access flags (see @link Opcodes{}).
 * @param name the method's name.
 * @param descriptor the method's descriptor (see @link Type{}).
 * @throws IllegalStateException if a subclass calls this constructor.
 */

Found in path(s):
* /opt/cola/permits/1135892456_1613626901.70/asm-commons-7-1-sources-2-
jar/org/objectweb/asm/commons/GeneratorAdapter.java
No license file was found, but licenses were detected in source scan.
All rights reserved.
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the copyright holders nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

Found in path(s):
* /opt/cola/permits/1135892456_1613626901.7/0/asm-commons-7-1-sources-2.jar/org/objectweb/asm/commons/JSRInlinerAdapter.java
* /opt/cola/permits/1135892456_1613626901.7/0/asm-commons-7-1-sources-2.jar/org/objectweb/asm/commons/Method.java
* /opt/cola/permits/1135892456_1613626901.7/0/asm-commons-7-1-sources-2.jar/org/objectweb/asm/commons/StaticInitMerger.java
* /opt/cola/permits/1135892456_1613626901.7/0/asm-commons-7-1-sources-2.jar/org/objectweb/asm/commons/ModuleResolutionAttribute.java
* /opt/cola/permits/1135892456_1613626901.7/0/asm-commons-7-1-sources-2.jar/org/objectweb/asm/commons/Remapper.java
* /opt/cola/permits/1135892456_1613626901.7/0/asm-commons-7-1-sources-2.jar/org/objectweb/asm/commons/TryCatchBlockSorter.java
* /opt/cola/permits/1135892456_1613626901.7/0/asm-commons-7-1-sources-2.jar/org/objectweb/asm/commons/MethodRemapper.java
* /opt/cola/permits/1135892456_1613626901.7/0/asm-commons-7-1-sources-2.jar/org/objectweb/asm/commons/AdviceAdapter.java
* /opt/cola/permits/1135892456_1613626901.7/0/asm-commons-7-1-sources-2.jar/org/objectweb/asm/commons/AnnotationRemapper.java
* /opt/cola/permits/1135892456_1613626901.7/0/asm-commons-7-1-sources-2.jar/org/objectweb/asm/commons/InstructionAdapter.java
* /opt/cola/permits/1135892456_1613626901.7/0/asm-commons-7-1-sources-2.jar/org/objectweb/asm/commons/ClassRemapper.java
* /opt/cola/permits/1135892456_1613626901.7/0/asm-commons-7-1-sources-2.jar/org/objectweb/asm/commons/ModuleRemapper.java
* /opt/cola/permits/1135892456_1613626901.7/0/asm-commons-7-1-sources-2.jar/org/objectweb/asm/commons/AnalyzerAdapter.java
* /opt/cola/permits/1135892456_1613626901.7/0/asm-commons-7-1-sources-2.jar/org/objectweb/asm/commons/ModuleTargetAttribute.java
* /opt/cola/permits/1135892456_1613626901.7/0/asm-commons-7-1-sources-2.jar/org/objectweb/asm/commons/ModuleHashesAttribute.java
* /opt/cola/permits/1135892456_1613626901.7/0/asm-commons-7-1-sources-2.jar/org/objectweb/asm/commons/TableSwitchGenerator.java
* /opt/cola/permits/1135892456_1613626901.7/0/asm-commons-7-1-sources-2.jar/org/objectweb/asm/commons/SimpleRemapper.java
* /opt/cola/permits/1135892456_1613626901.7/0/asm-commons-7-1-sources-2.jar
jar/org/objectweb/asm/commons/FieldRemapper.java
*/opt/cola/permits/1135892456_1613626901.7/0/asm-commons-7-1-sources-2-jar/org/objectweb/asm/commons/CodeSizeEvaluator.java
*/opt/cola/permits/1135892456_1613626901.7/0/asm-commons-7-1-sources-2-jar/org/objectweb/asm/commons/SignatureRemapper.java
No license file was found, but licenses were detected in source scan.

2011 INRIA, France Telecom
* All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in the
* documentation and/or other materials provided with the distribution.
* 3. Neither the name of the copyright holders nor the names of its
* contributors may be used to endorse or promote products derived from
* this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
* LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
* CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
* SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
* INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
* CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
* THE POSSIBILITY OF SUCH DAMAGE.

Found in path(s):
*/opt/cola/permits/1135892456_1613626901.7/0/asm-commons-7-1-sources-2-jar/org/objectweb/asm/commons/package.html
No license file was found, but licenses were detected in source scan.

// All rights reserved.
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions
// are met:
// 1. Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
// 2. Redistributions in binary form must reproduce the above copyright
// notice, this list of conditions and the following disclaimer in the
// documentation and/or other materials provided with the distribution.
// 3. Neither the name of the copyright holders nor the names of its
// this software without specific prior written permission.
/**
 * Constructs a new {@link SerialVersionUIDAdder}. <i>Subclasses must not use this
 * constructor</i>. Instead, they must use the {@link #SerialVersionUIDAdder(int, ClassVisitor)}
 * version.
 * @param classVisitor a {@link ClassVisitor} to which this visitor will delegate calls.
 * @throws IllegalStateException If a subclass calls this constructor.
 */

1.304 java-driver-core 4.10.0
1.304.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright DataStax, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

// under the Apache License, Version 2.0.
// The ASF licenses this file to You under the Apache License, Version 2.0
// (the "License"); you may not use this file except in compliance with
// the License. You may obtain a copy of the License at
// http://www.apache.org/licenses/LICENSE-2.0
// distributed under the License is distributed on an "AS IS" BASIS,

Found in path(s):
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-
jar/com/datastax/dse/driver/internal/core/search/DateRangeUtil.java

No license file was found, but licenses were detected in source scan.
/*
 * Copyright DataStax, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

/* The remainder of this class is largely based on jnr.ffi.Platform in jnr-ffi version 2.1.10.
 * We copy it manually here in order to avoid introducing an extra dependency merely for the sake of
 * evaluating some system properties.
 *
 * jnr-ffi copyright notice follows:
 *
 * Copyright (C) 2008-2010 Wayne Meissner
 *
 * This file is part of the JNR project.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/oss/driver/internal/core/os/CpuInfo.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright DataStax, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 */
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
* /

Found in path(s):
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-
  jar/com/datastax/oss/driver/internal/core/metadata/token/NetworkTopologyReplicationStrategy.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-
  jar/com/datastax/oss/driver/internal/core/metadata/token/KeyspaceTokenMap.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-
  jar/com/datastax/oss/driver/internal/core/addresstranslation/PassThroughAddressTranslator.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-
  jar/com/datastax/oss/driver/internal/core/time/AtomicTimestampGenerator.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-
  jar/com/datastax/dse/driver/internal/core/metadata/schema/DefaultDseViewMetadata.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-
  jar/com/datastax/oss/driver/api/core/config/OptionsMap.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-
  jar/com/datastax/oss/driver/api/core/session/SessionBuilder.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-
  jar/com/datastax/oss/driver/internal/core/metadata/schema/queries/DefaultSchemaQueriesFactory.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-
  jar/com/datastax/oss/driver/api/core/connection/package-info.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-
  jar/com/datastax/oss/driver/internal/core/loadbalancing/nodeset/MultiDcNodeSet.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-
  jar/com/datastax/oss/driver/api/core/servererrors/SyntaxException.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-
  jar/com/datastax/oss/driver/internal/core/metadata/DefaultMetadata.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/oss/driver/internal/core/metadata/token/ByteOrderedToken.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/oss/driver/internal/core/metadata/CloudTopologyMonitor.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/oss/driver/internal/core/session/throttling/PassThroughRequestThrottler.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/oss/driver/internal/core/metadata/schema/parsing/SchemaParserFactory.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/oss/driver/api/core/ssl/SslEngineFactory.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/dse/driver/internal/core/cql/reactive/FailedReactiveResultSet.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/oss/driver/api/core/metadata/SafeInitNodeStateListener.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/oss/driver/api/core/metadata/schema/parsing/SimpleJsonParser.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/oss/driver/api/core/metadata/schema/events/FunctionChangeEvent.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/dse/driver/internal/core/cql/continuous/reactive/ContinuousCqlRequestReactiveProcessor.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/dse/driver/internal/core/cql/continuous/ContinuousCqlRequestSyncProcessor.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/oss/driver/api/core/metadata/schema/IndexMetadata.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/dse/driver/internal/core/cql/continuous/reactive/ContinuousCqlRequestReactiveProcessor.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/oss/driver/api/core/metadata/schema/FunctionMetadata.java

Open Source Used In security_automation identity-guest-cs 2810
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/internal/core/type/codec/TimestampCodec.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/internal/core/config/DerivedExecutionProfile.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/internal/core/protocol/Lz4Compressor.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/internal/core/metadata/schema/queries/CassandraSchemaQueries.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/dse/driver/internal/core/graph/GraphProtocol.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/internal/core/metrics/HdrReservoir.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/api/core/type/reflect/GenericTypeParameter.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/internal/core/metadata/schema/queries/SchemaQueriesFactory.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/dse/driver/internal/core/graph/ContinuousAsyncGraphResultSet.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/dse/driver/api/core/cql/reactive/ReactiveSession.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/api/core/ProtocolVersion.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/api/core/session/ProgrammaticArguments.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/api/core/servererrors/DefaultWriteType.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/api/ssl/package-info.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/api/core/session/ProgrammaticArguments.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/api/core/session/ProgrammaticArguments.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/dse/driver/internal/core/graph/binary/EditDistanceSerializer.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/api/decrypt/AttachmentPoint.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/api/core/cql/ExecutionInfo.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/api/core/type/codec/SimpleBlobCodec.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/api/core/servererrors/WriteTimeoutException.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/api/core/auth/ProgrammaticPlainTextAuthProvider.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/api/core/data/GettableById.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/api/core/graph/reactive/ReactiveGraphResultSet.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/api/core/metadata/TokenMap.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/internal/core/servererrors/DefaultWriteTypeRegistry.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/api/core/adminrequest/AdminRequestHandler.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/api/core/loadbalancing/LoadBalancingPolicy.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/api/core/metadata/schema/DseViewMetadata.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/api/core/util/concurrent/CycleDetector.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/api/core/session/SessionWrapper.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/api/core/servererrors/FunctionFailureException.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/internal/core/codec/GeometryCodec.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/internal/core/cql/reactive/SimpleUnicastProcessor.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/api/core/type/DataType.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/internal/core/context/DefaultNettyOptions.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/api/data/CqlDuration.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/dse/driver/internal/core/graph/EditDistance.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/api/core/config/composite/CompositeDriverConfig.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/api/core/type/codec/BlobCodec.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/api/core/type/codec/PrimitiveLongCodec.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/api/core/type/codec/PrimitiveIntCodec.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/dse/driver/internal/core/loadbalancing/DseLoadBalancingPolicy.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/dse/driver/api/core/graph/TinkerpopBufferUtil.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/api/core/AsyncAutoCloseable.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/api/core/CqlSessionBuilder.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/api/core/config/DriverConfig.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/api/core/config/DriverConfigLoader.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/api/core/cql/continuous/reactive/DefaultContinuousReactiveResultSet.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/dse/driver/api/core/graph/FluentGraphStatementBuilder.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/dse/driver/api/core/config/DseDriverConfigLoader.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/api/core/cql/StatementBuilder.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/api/core/config/DriverConfig.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/api/core/context/NettyOptions.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/api/core/auth/AuthProvider.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/api/core/type/codec/geometry/LineStringCodec.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/api/core/cql/CqlPrepareHandler.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/dse/driver/internal/core/graph/binary/buffer/DseNettyBufferFactory.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/oss/driver/internal/core/metrics/DropwizardSessionMetricUpdater.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/oss/driver/internal/core/session/SchemaListenerNotifier.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/dse/driver/api/core/cql/reactive/ReactiveRow.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/oss/driver/internal/core/metadata/schema/queries/SchemaQueries.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/oss/driver/internal/core/metadata/schema/DefaultKeyspaceMetadata.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/oss/driver/api/core/metrics/SessionMetric.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/oss/driver/api/core/data/time/Clock.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/oss/driver/api/core/data/time/DateRangePrecision.java
Open Source Used In security_automation identity-guest-cs

* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/oss/driver/internal/core/ConsistencyLevelRegistry.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/oss/driver/internal/core/metadata/schema/queries/SchemaRows.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/oss/driver/internal/core/metadata/schema/ScriptBuilder.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/oss/driver/internal/core/metadata/schema/ViewMetadata.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/oss/driver/internal/core/metrics/NodeMetric.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/dse/driver/internal/core/cql/reactive/EmptySubscription.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/dse/driver/internal/core/cql/reactive/FailedReactiveGraphResultSet.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/oss/driver/internal/core/protocol/BytesToSegmentDecoder.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/oss/driver/internal/core/servererrors/WriteTypeRegistry.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/oss/driver/internal/core/time/NativeClock.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/oss/driver/api/core/data/UdtValue.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/oss/driver/api/core/type/CustomType.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/dse/driver/api/core/graph/GraphStatementBuilderBase.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/dse/driver/internal/core/graph/GraphRequestHandler.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/internal/core/metadata/schema/queries/KeyspaceFilter.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/internal/core/type/codec/extras/OptionalCodec.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/internal/core/type/codec/extras/array/DoubleListToArrayCodec.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/dse/driver/api/core/cql/reactive/ReactiveQueryMetadata.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/internal/core/cql/DseConversions.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/api/core/retry/RetryPolicy.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/api/core/config/DriverExecutionProfile.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/api/core/servererrors/UnauthorizedException.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/api/core/cql/DefaultTraceEvent.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/api/core/connection/FrameTooLongException.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/api/core/metadata/schema/parsing/AggregateParser.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/api/core/type/codec/TypeCodecs.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/api/core/data/SettableById.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/api/core/data/TupleValue.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/internal/core/codec/ListCodec.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/INTERNAL/core/cql/DefaultBatchStatement.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/INTERNAL/core/metadata/LoadBalancingPolicyWrapper.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/dse/driver/INTERNAL/core/data/geometry/DefaultPoint.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/dse/driver/api/core/auth/DsePlainTextAuthProvider.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/dse/driver/api/core/metadata/schema/DseKeyspaceMetadata.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/dse/driver/INTERNAL/core/channel/InFlightHandler.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/INTERNAL/core/os/JnrLibc.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/INTERNAL/core/adminrequest/AdminResult.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/INTERNAL/core/config/typesafe/DefaultDriverConfigLoader.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/dse/driver/INTERNAL/core/insights/schema/InsightsStatusData.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/dse/driver/INTERNAL/core/metadata/schema/parsing/RelationParser.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/dse/driver/INTERNAL/core/insights/ExecutionProfilesInfoFinder.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/INTERNAL/core/insights/InsightsClient.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/INTERNAL/core/type/UserDefinedTypeBuilder.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/INTERNAL/core/metadata/schema/parsing/DataTypeParser.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/dse/driver/INTERNAL/core/DseProtocolFeature.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/INTERNAL/core/codec/UdtCodec.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/INTERNAL/package-info.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/oss/driver/internal/core/loadbalancing/nodeset/NodeSet.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/oss/driver/internal/core/util/CollectionsUtils.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/dse/driver/internal/core/cql/continuous/DefaultContinuousAsyncResultSet.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/oss/driver/internal/core/util/concurrent/PromiseCombiner.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/oss/driver/internal/core/metadata/schema/parsing/DseAggregateParser.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/dse/driver/internal/core/metadata/schema/parsing/RuleBasedKeyspaceFilter.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/oss/driver/internal/core/metadata/token/TokenFactoryRegistry.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/dse/driver/internal/core/cql/continuous/DefaultContinuousAsyncResultSet.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/dse/driver/internal/core/metadata/schema/parsing/RawColumn.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/oss/driver/internal/core/metadata/token/TokenFactoryRegistry.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/oss/driver/api/core/NoNodeAvailableException.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/oss/driver/api/core/Session.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/oss/driver/api/core/session/Request.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/oss/driver/internal/metadata/schema/parsing/DataTypeClassNameCompositeParser.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/oss/driver/api/core/CqlSession.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/oss/driver/internal/core/loadbalancing/helper/InferringLocalDcHelper.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/dse/driver/internal/core/graph/DefaultFluentGraphStatement.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/oss/driver/internal/core/metadata/schema/ShallowUserDefinedType.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/oss/driver/internal/core/metadata/RemoveNodeRefresh.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/dse/driver/internal/core/graph/binary/UdtValueSerializer.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/oss/driver/internal/core/metadata/schema/parsing/SchemaParser.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/oss/driver/api/core/retry/RetryDecision.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/oss/driver/internal/core/adminrequest/ThrottledAdminRequestHandler.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/oss/driver/api/core/config/ProgrammaticDriverConfigLoaderBuilder.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/oss/driver/api/core/MavenCoordinates.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/oss/driver/internal/core/metrics/MetricPaths.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/oss/driver/internal/core/metadata/schema/parsing/SchemaParser.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/oss/driver/api/retry/RetryDecision.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/oss/driver/internal/core/metadata/schema/DefaultDseFunctionMetadata.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/oss/driver/internal/core/cql/reactive/ReactiveResultSetSubscription.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/oss/driver/internal/core/metadata/NodeStateManager.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/oss/driver/dse/driver/internal/core/metadata/schema/DefaultDseFunctionMetadata.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/oss/driver/dse/driver/internal/core/insights/schema/SpecificExecutionProfile.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/oss/driver/internal/core/cql/reactive/ReactiveResultSetSubscription.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/oss/driver/dse/driver/internal/core/graph/GraphExecutionInfoConverter.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/oss/driver/dse/driver/internal/core/loadbalancing/DefaultLoadBalancingPolicy.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/oss/driver/api/core/cql/DefaultPreparedStatement.java
Open Source Used In security_automation identity-guest-cs 2834
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/internal/core/metadata/schema/events/AggregateChangeEvent.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/internal/core/util/DependencyCheck.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/dse/driver/internal/core/tracker/MultiplexingRequestTracker.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/dse/driver/api/core/metadata/schema/DseFunctionMetadata.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/internal/core/util/concurrent/CompletableFutures.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/internal/core/util/collection/LazyQueryPlan.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/dse/driver/internal/core/graph/reactive/ReactiveGraphResultSetSubscription.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/dse/driver/api/core/loadbalancing/DseDcInferringLoadBalancingPolicy.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/dse/driver/api/core/metadata/schema/parsing/FunctionParser.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/dse/driver/api/core/graph/predicates/CqlCollection.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/dse/driver/internal/core/graph/binary/GeometrySerializer.java
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1.jar/com/datastax/oss/driver/internal/core/os/Libc.java
No license file was found, but licenses were detected in source scan.

Copyright DataStax, Inc.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and limitations under the License.

Found in path(s):
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/META-INF/maven/com.datastax.oss/java-driver-core/pom.xml

No license file was found, but licenses were detected in source scan.

/
* Copyright DataStax, Inc.
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
/**
* Returns a gateway to the driver's DropWizard metrics, or {@link Optional#empty()} if all
* metrics are disabled, or if the driver has been configured to use MicroProfile or Micrometer
* instead of DropWizard (see {@code advanced.metrics.factory.class} in the configuration).
*<p>{@link Metrics} was originally intended to allow programmatic access to the metrics, but it
* has a hard dependency to the DropWizard API, which makes it unsuitable for alternative metric
* frameworks. A workaround is to inject your own metric registry with {@link SessionBuilder#withMetricRegistry(Object)} when you build the session. You can then use the
* framework's proprietary APIs to retrieve the metrics from the registry.
*/

Found in path(s):
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-jar/com/datastax/oss/driver/api/core/session/Session.java

No license file was found, but licenses were detected in source scan.

# Licensed under the Apache License, Version 2.0 (the "License");
# you may not use this file except in compliance with the License.
# You may obtain a copy of the License at
# http://www.apache.org/licenses/LICENSE-2.0
# distributed under the License is distributed on an "AS IS" BASIS,

Found in path(s):
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-
jar/com/datastax/oss/driver/Driver.properties
No license file was found, but licenses were detected in source scan.

/*! 
 * Copyright DataStax, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
/**
 * Prepares a CQL statement asynchronously (the call returns as soon as the prepare query was
 * sent, generally before the statement is prepared).
 *
 * Note that the bound statements created from the resulting prepared statement will inherit
 * some of the attributes of {code query}; see {link SyncCqlSession#prepare(SimpleStatement)}
 * for more details.
 *
 * The result of this method is cached (see {link SyncCqlSession#prepare(SimpleStatement)} for
 * more explanations).
 *
 * @param statement the CQL query to prepare (that can be any {code SimpleStatement}).
 * @return a {code CompletionStage} that, once complete, will produce the prepared statement.
 */

Found in path(s):
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-
  jar/com/datastax/oss/driver/api/core/cql/AsyncCqlSession.java
No license file was found, but licenses were detected in source scan.

/*! 
 * Copyright DataStax, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

// Copyright 2008 Google Inc. All rights reserved.
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions are
// * Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
// * Redistributions in binary form must reproduce the above
// copyright notice, this list of conditions and the following disclaimer
// in the documentation and/or other materials provided with the
// * Neither the name of Google Inc. nor the names of its
// this software without specific prior written permission.

Found in path(s):
* /opt/cola/permits/1199743879_1630577640.01/0/java-driver-core-4-10-0-sources-1-
jar/com/datastax/oss/driver/internal/core/type/util/VIntCoding.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright DataStax, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

/*
* which was originally released under the Apache 2.0 license with the following copyright:
*
* Licensed to the Apache Software Foundation (ASF) under one or more
* contributor license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright ownership.
* The ASF licenses this file to You under the Apache License, Version 2.0
* (the "License"); you may not use this file except in compliance with
* the License. You may obtain a copy of the License at
*
http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.305 spring-cloud-starter-consul 3.0.4
1.305.1 Available under license:

No license file was found, but licenses were detected in source scan.

<name>Apache License, Version 2.0</name>
Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at distributed under the License is distributed on an "AS IS" BASIS,

1.306 latencyutils 2.0.3

1.307 ojdbc 19.3.0.0
1.307.1 Available under license:

No license file was found, but licenses were detected in source scan.

All rights reserved.-->

# or the Oracle proprietary type oracle.xdb.XMLType.

No license file was found, but licenses were detected in source scan.

# or the Oracle proprietary type oracle.xdb.XMLType.
1.308 zstd-jni 1.4.5-6

1.308.1 Available under license:

BSD License

For Zstandard software

Copyright (c) 2016-present, Facebook, Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name Facebook nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Zstd-jni: JNI bindings to Zstd Library

Copyright (c) 2015-present, Luben Karavelov/ All rights reserved.

BSD License

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this
list of conditions and the following disclaimer in the documentation and/or
other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR
ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON
ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
License is intended to guarantee your freedom to share and change free
software--to make sure the software is free for all its users. This
General Public License applies to most of the Free Software
Foundation's software and to any other program whose authors commit to
using it. (Some other Free Software Foundation software is covered by
the GNU Lesser General Public License instead.) You can apply it to
your programs, too.

When we speak of free software, we are referring to freedom, not
price. Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for
this service if you wish), that you receive source code or can get it
if you want it, that you can change the software or use pieces of it
in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid
anyone to deny you these rights or to ask you to surrender the rights.
These restrictions translate to certain responsibilities for you if you
distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether
gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate
copyright notice and disclaimer of warranty; keep intact all the
notices that refer to this License and to the absence of any warranty;
and give any other recipients of the Program a copy of this License
along with the Program.

You may charge a fee for the physical act of transferring a copy, and
you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion
of it, thus forming a work based on the Program, and copy and
distribute such modifications or work under the terms of Section 1
above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices
stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in
whole or in part contains or is derived from the Program or any
part thereof, to be licensed as a whole at no charge to all third
parties under the terms of this License.

c) If the modified program normally reads commands interactively
when run, you must cause it, when started running for such
interactive use in the most ordinary way, to print or display an
announcement including an appropriate copyright notice and a
notice that there is no warranty (or else, saying that you provide
a warranty) and that users may redistribute the program under
these conditions, and telling the user how to view a copy of this
License. (Exception: if the Program itself is interactive but
does not normally print such an announcement, your work based on
the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If
identifiable sections of that work are not derived from the Program,
and can be reasonably considered independent and separate works in
themselves, then this License, and its terms, do not apply to those
sections when you distribute them as separate works. But when you
distribute the same sections as part of a whole which is a work based
on the Program, the distribution of the whole must be on the terms of
this License, whose permissions for other licensees extend to the
entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest
your rights to work written entirely by you; rather, the intent is to
exercise the right to control the distribution of derivative or
collective works based on the Program.

In addition, mere aggregation of another work not based on the Program
3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such
parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.
This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING
OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED
TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY
YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER
PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE
POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest
possible use to the public, the best way to achieve this is to make it
free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest
to attach them to the start of each source file to most effectively
convey the exclusion of warranty; and each file should have at least
the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year>  <name of author>

This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License along
with this program; if not, write to the Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this
when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate
parts of the General Public License. Of course, the commands you use may
be called something other than `show w' and `show c'; they could even be
mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the program, if
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program
`Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into
proprietary programs. If your program is a subroutine library, you may
consider it more useful to permit linking proprietary applications with the
library. If this is what you want to do, use the GNU Lesser General
Public License instead of this License.

1.309 commons-compiler 3.1.6
1.309.1 Available under license:
No license file was found, but licenses were detected in source scan.

/*
 * Janino - An embedded Java[TM] compiler
 * 
 * Copyright (c) 2018 Arno Unkrig. All rights reserved.
 * 
 * Redistribution and use in source and binary forms, with or without modification, are permitted provided that the
 * following conditions are met:
 * 
 * 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the
 * following disclaimer.
 * 
 * 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the
 * following disclaimer in the documentation and/or other materials provided with the distribution.
 * 
 * 3. Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote
 * products derived from this software without specific prior written permission.
 * 
 * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
 * ANY EXPRESS OR IMPLIED WARRANTIES,
 * INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND
 * FITNESS FOR A PARTICULAR PURPOSE ARE
 * DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR
 * ANY DIRECT, INDIRECT, INCIDENTAL,
 * SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
 * PROCUREMENT OF SUBSTITUTE GOODS OR
* SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED
AND ON ANY THEORY OF LIABILITY,
* WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR
OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
* OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

Found in path(s):
* /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-
jar/org/codehaus/commons/compiler/util/package-info.java
* /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-
jar/org/codehaus/commons/compiler/java8/java/util/function/Consumer.java
* /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-
jar/org/codehaus/commons/compiler/util/Disassembler.java
* /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-
jar/org/codehaus/commons/compiler/java9/java/lang/module/ModuleReader.java
* /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-
jar/org/codehaus/commons/compiler/java9/java/lang/module/ModuleFinder.java
* /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-
jar/org/codehaus/commons/compiler/java8/java/util/stream/Stream.java
* /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-
jar/org/codehaus/commons/compiler/java9/java/lang/module/ModuleReference.java

No license file was found, but licenses were detected in source scan.

/*
 * Janino - An embedded Java(TM) compiler
 *
 * Copyright (c) 2001-2010 Arno Unkrig. All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without modification, are permitted provided that the
 * following conditions are met:
 *
 * 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the
 *    following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the
 *    following disclaimer in the documentation and/or other materials provided with the distribution.
 * 3. Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote
 *    products derived from this software without specific prior written permission.
 *
 * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
 * ANY EXPRESS OR IMPLIED WARRANTIES,
 * INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND
 * FITNESS FOR A PARTICULAR PURPOSE ARE
 * DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR
 * ANY DIRECT, INDIRECT, INCIDENTAL,
 * SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/

Found in path(s):
* /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-jar/org/codehaus/commons/compiler/IExpressionEvaluator.java
* /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-jar/org/codehaus/commons/compiler/util/StringPattern.java
* /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-jar/org/codehaus/commons/compiler/util/DirectoryIterator.java
* /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-jar/org/codehaus/commons/compiler/util/resource/Resource.java
* /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-jar/org/codehaus/commons/compiler/samples/ClassBodyDemo.java
* /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-jar/org/codehaus/commons/compiler/LocatedException.java
* /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-jar/org/codehaus/commons/compiler/WarningHandler.java
* /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-jar/org/codehaus/commons/compiler/IScriptEvaluator.java
* /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-jar/org/codehaus/commons/compiler/util/iterator/UniterableElementException.java
* /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-jar/org/codehaus/commons/compiler/util/iterator/EnumerationIterator.java
* /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-jar/org/codehaus/commons/compiler/util/iterator/IteratorCollection.java
* /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-jar/org/codehaus/commons/compiler/util/resource/ResourceFinder.java
* /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-jar/org/codehaus/commons/compiler/util/resource/MultiResourceFinder.java
* /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-jar/org/codehaus/commons/compiler/util/reflect/ByteArrayClassLoader.java
* /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-jar/org/codehaus/commons/compiler/ICookable.java
* /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-jar/org/codehaus/commons/compiler/util/IteratorCollection.java
* /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-jar/org/codehaus/commons/compiler/util/resource/ResourceFinder.java
* /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-jar/org/codehaus/commons/compiler/util/resource/MultiResourceFinder.java
* /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-jar/org/codehaus/commons/compiler/util/resource/ResourceFinder.java
* /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-jar/org/codehaus/commons/compiler/util/resource/MultiResourceFinder.java
* /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-jar/org/codehaus/commons/compiler/util/resource/ResourceFinder.java
* /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-jar/org/codehaus/commons/compiler/util/resource/MultiResourceFinder.java
* /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-jar/org/codehaus/commons/compiler/util/resource/ResourceFinder.java
* /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-jar/org/codehaus/commons/compiler/util/resource/MultiResourceFinder.java
/*
* Janino - An embedded Java[TM] compiler
*
* Copyright (c) 2013 Arno Unkrig. All rights reserved.
* 
* Redistribution and use in source and binary forms, with or without modification, are permitted provided that the
* following conditions are met:
* 
* 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the
*    following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the
*    following disclaimer in the documentation and/or other materials provided with the distribution.
* 3. Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote
*    products derived from this software without specific prior written permission.
* 
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
* ANY EXPRESS OR IMPLIED WARRANTIES,
* INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND
* FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR
* ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
* PROCUREMENT OF SUBSTITUTE GOODS OR
* SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED
* AND ON ANY THEORY OF LIABILITY,
* WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR
* OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
* OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/
No license file was found, but licenses were detected in source scan.

/*
 * Janino - An embedded Java[TM] compiler
 *
 * Copyright (c) 2010 Arno Unkrig. All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without modification, are permitted provided that the
 * following conditions are met:
 *
 * 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the
 *    following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the
 *    following disclaimer in the documentation and/or other materials provided with the distribution.
 * 3. Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote
 *    products derived from this software without specific prior written permission.
 *
 * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
 * ANY EXPRESS OR IMPLIED WARRANTIES,
 * INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND
 * FITNESS FOR A PARTICULAR PURPOSE ARE
 * DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR
 * ANY DIRECT, INDIRECT, INCIDENTAL,
 * SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
 * PROCUREMENT OF SUBSTITUTE GOODS OR
 * SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED
 * AND ON ANY THEORY OF LIABILITY,
 * WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR
 * OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
 * OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
 */

No license file was found, but licenses were detected in source scan.

/*
 * Janino - An embedded Java[TM] compiler
 *
 * Copyright (c) 2016 Arno Unkrig. All rights reserved.
 *
* Redistribution and use in source and binary forms, with or without modification, are permitted provided that the
  * following conditions are met:
  *
  * 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the
     * following disclaimer.
  * 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the
     * following disclaimer in the documentation and/or other materials provided with the distribution.
  * 3. Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote
     * products derived from this software without specific prior written permission.
  *
  * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
  * ANY EXPRESS OR IMPLIED WARRANTIES,
  * INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND
  * FITNESS FOR A PARTICULAR PURPOSE ARE
  * DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR
  * ANY DIRECT, INDIRECT, INCIDENTAL,
  * SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
  * PROCUREMENT OF SUBSTITUTE GOODS OR
  * SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED
  * AND ON ANY THEORY OF LIABILITY,
  * WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR
  * OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
  * OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
  */

Found in path(s):
  * /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-jar/org/codehaus/commons/nullanalysis/Nullable.java
  * /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-jar/org/codehaus/commons/nullanalysis/NotNull.java
  * /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-jar/org/codehaus/commons/nullanalysis/package-info.java
  * /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-jar/org/codehaus/commons/nullanalysis/NotNullByDefault.java

No license file was found, but licenses were detected in source scan.

/*
 * Janino - An embedded Java[TM] compiler
 *
 * Copyright (c) 2018, 2019 Arno Unkrig. All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without modification, are permitted provided that the
 * following conditions are met:
 *
 * 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the
    following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the
    following disclaimer in the documentation and/or other materials provided with the distribution.
 */
3. Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/

Found in path(s):
* /opt/cola/permits/1199651225_1630558002.90/commons-compiler-3-1-6-sources.jar/org/codehaus/commons/compiler/io/Readers.java

No license file was found, but licenses were detected in source scan.

/*
 * Janino - An embedded Java[TM] compiler
 *
 * Copyright (c) 2020 Arno Unkrig. All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
 *
 * 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
 * 3. Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.
 *
 * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY,
* WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR
OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
* OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

Found in path(s):
* /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-
jar/org/codehaus/commons/compiler/util/SystemProperties.java
No license file was found, but licenses were detected in source scan.

*/
* Janino - An embedded Java[TM] compiler
*
* Copyright (c) 2001-2017 Arno Unkrig. All rights reserved.
*
* Redistribution and use in source and binary forms, with or without modification, are permitted provided that the
* following conditions are met:
*
* 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the
   following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the
   following disclaimer in the documentation and/or other materials provided with the distribution.
* 3. Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote
   products derived from this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES,
* INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND
FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR
ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
PROCUREMENT OF SUBSTITUTE GOODS OR
* SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED
AND ON ANY THEORY OF LIABILITY,
* WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR
OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
* OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

Found in path(s):
* /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-
jar/org/codehaus/commons/compiler/Sandbox.java
No license file was found, but licenses were detected in source scan.

*/
* Janino - An embedded Java[TM] compiler
*
* Copyright (c) 2019 Arno Unkrig. All rights reserved.

* Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* 3. Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/

Found in path(s):
* /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-jar/org/codehaus/commons/compiler/ICompiler.java
* /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-jar/org/codehaus/commons/compiler/lang/package-info.java
* /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-jar/org/codehaus/commons/compiler/util/Predicate.java
* /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-jar/org/codehaus/commons/compiler/util/reflect/ApiLog.java
* /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-jar/org/codehaus/commons/compiler/util/resource/ResourceFinders.java
* /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-jar/org/codehaus/commons/compiler/util/resource/LocatableResource.java
* /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-jar/org/codehaus/commons/compiler/lang/ClassLoaders.java
* /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-jar/org/codehaus/commons/compiler/util/reflect/Methods.java
* /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-jar/org/codehaus/commons/compiler/samples/CompilerDemo.java
* /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-jar/org/codehaus/commons/compiler/util/StringUtil.java
* /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-jar/org/codehaus/commons/compiler/IMultiCookable.java
* /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-jar/org/codehaus/commons/compiler/java8/java/util/package-info.java
* /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-jar/org/codehaus/commons/compiler/util/reflect/package-info.java
* /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-jar/org/codehaus/commons/compiler/MultiCookable.java
* /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-jar/org/codehaus/commons/compiler/util/iterator/Iterables.java
* /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-jar/org/codehaus/commons/compiler/io/package-info.java
* /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-jar/org/codehaus/commons/compiler/io/InputStreams.java
* /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-jar/org/codehaus/commons/compiler/util/reflect/Classes.java
* /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-jar/org/codehaus/commons/compiler/util/Numbers.java
* /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-jar/org/codehaus/commons/compiler/java8/java/util/function/package-info.java
* /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-jar/org/codehaus/commons/compiler/util/reflect/Proxies.java
* /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-jar/org/codehaus/commons/compiler/java8/java/util/function/Consumers.java
* /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-jar/org/codehaus/commons/compiler/util/iterator/package-info.java
* /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-jar/org/codehaus/commons/compiler/java8/java/util/stream/package-info.java
* /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-jar/org/codehaus/commons/compiler/util/resource/ListableResourceFinder.java
* /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-jar/org/codehaus/commons/compiler/util/reflect/NoException.java
* /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-jar/org/codehaus/commons/compiler/io/package-info.java

No license file was found, but licenses were detected in source scan.

/*
 * Janino - An embedded Java[TM] compiler
 *
 * Copyright (c) 2001-2018 Arno Unkrig. All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without modification, are permitted provided that the
 * following conditions are met:
 *
 * 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the
 * following disclaimer.
 */
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/

Found in path(s):
* /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-jar/org/codehaus/commons/compiler/util/LineAndColumnTracker.java
* /opt/cola/permits/1199651225_1630558002.9/0/commons-compiler-3-1-6-sources-jar/org/codehaus/commons/compiler/util/resource/StringResource.java

1.310 netty-codec-dns 4.1.69.Final

1.310.1 Available under license:
No license file was found, but licenses were detected in source scan.

*/
* Copyright 2021 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License, version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at:
*
*    https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
*/
Open Source Used In security_automation identity-guest-cs 2865

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2015 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
*  https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

/*
* Found in path(s):
* /opt/cola/permits/1221869446_1635493924.46/0/netty-codec-dns-4-1-69-final-sources-
* jar/io/netty/handler/codec/dns/TcpDnsResponseEncoder.java
* /opt/cola/permits/1221869446_1635493924.46/0/netty-codec-dns-4-1-69-final-sources-
* jar/io/netty/handler/codec/dns/TcpDnsQueryDecoder.java
* /opt/cola/permits/1221869446_1635493924.46/0/netty-codec-dns-4-1-69-final-sources-
* jar/io/netty/handler/codec/dns/AbstractDnsMessage.java
* /opt/cola/permits/1221869446_1635493924.46/0/netty-codec-dns-4-1-69-final-sources-
* jar/io/netty/handler/codec/dns/DnsMessageUtil.java
* /opt/cola/permits/1221869446_1635493924.46/0/netty-codec-dns-4-1-69-final-sources-
* jar/io/netty/handler/codec/dns/DnsQuery.java
* /opt/cola/permits/1221869446_1635493924.46/0/netty-codec-dns-4-1-69-final-sources-
* jar/io/netty/handler/codec/dns/DefaultDnsRawRecord.java
* /opt/cola/permits/1221869446_1635493924.46/0/netty-codec-dns-4-1-69-final-sources-
* jar/io/netty/handler/codec/dns/DatagramDnsResponseEncoder.java
* /opt/cola/permits/1221869446_1635493924.46/0/netty-codec-dns-4-1-69-final-sources-
* jar/io/netty/handler/codec/dns/DatagramDnsQueryDecoder.java
*/
*/opt/cola/permits/1221869446_1635493924.46/0/netty-codec-dns-4-1-69-final-sources-jar/io/netty/handler/codec/dns/AbstractDnsRecord.java
*/opt/cola/permits/1221869446_1635493924.46/0/netty-codec-dns-4-1-69-final-sources-jar/io/netty/handler/codec/dns/DnsRecordDecoder.java
*/opt/cola/permits/1221869446_1635493924.46/0/netty-codec-dns-4-1-69-final-sources-jar/io/netty/handler/codec/dns/DnsSection.java
*/opt/cola/permits/1221869446_1635493924.46/0/netty-codec-dns-4-1-69-final-sources-jar/io/netty/handler/codec/dns/DnsRecord.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2016 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */
Found in path(s):
* /opt/cola/permits/1221869446_1635493924.46/0/netty-codec-dns-4-1-69-final-sources-jar/io/netty/handler/codec/dns/DefaultDnsPtrRecord.java
* /opt/cola/permits/1221869446_1635493924.46/0/netty-codec-dns-4-1-69-final-sources-jar/io/netty/handler/codec/dns/DnsPtrRecord.java

No license file was found, but licenses were detected in source scan.

~/ Copyright 2014 The Netty Project
~/

~/ The Netty Project licenses this file to you under the Apache License,
~/ version 2.0 (the "License"); you may not use this file except in compliance
~/ with the License. You may obtain a copy of the License at:
~/
~/ https://www.apache.org/licenses/LICENSE-2.0

~/ Unless required by applicable law or agreed to in writing, software
~/ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~/ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~/ License for the specific language governing permissions and limitations
~/ under the License.

Found in path(s):

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2019 The Netty Project
* 
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* 
* https://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
1.311 amqp-client 5.10.0

1.311.1 Available under license :

Found license 'General Public License 2.0' in '/* Copyright (c) 2007-2020 VMware, Inc. or its affiliates. All rights reserved. // This software, the RabbitMQ Java client library, is triple-licensed under the // Mozilla Public License 2.0 ("MPL"), the GNU General Public License version 2 ("GPL") and the Apache License version 2 ("ASL"). For the MPL, please see'
MPL, please see * more details.'
Found license 'General Public License 2.0' in '// Copyright (c) 2007-2020 VMware, Inc. or its affiliates. All rights reserved. // This software, the RabbitMQ Java client library, is triple-licensed under the // Mozilla Public License 2.0 ("MPL"), the GNU General Public License version 2 // ("GPL") and the Apache License version 2 ("ASL"). For the MPL, please see final static String LICENSE="Licensed under the MPL. See https://www.rabbitmq.com/";'
Found license 'General Public License 2.0' in '// Copyright (c) 2017-2020 VMware, Inc. or its affiliates. All rights reserved. // This software, the RabbitMQ Java client library, is triple-licensed under the // Mozilla Public License 2.0 ("MPL"), the GNU General Public License version 2 // ("GPL") and the Apache License version 2 ("ASL"). For the MPL, please see'
Found license 'General Public License 2.0' in '// Copyright (c) 2019-2020 VMware, Inc. or its affiliates. All rights reserved. // This software, the RabbitMQ Java client library, is triple-licensed under the // Mozilla Public License 2.0 ("MPL"), the GNU General Public License version 2 // ("GPL") and the Apache License version 2 ("ASL"). For the MPL, please see'
Found license 'General Public License 2.0' in '// Copyright (c) 2007-2020 VMware, Inc. or its affiliates. All rights reserved. // This software, the RabbitMQ Java client library, is triple-licensed under the // Mozilla Public License 1.1 ("MPL"), the GNU General Public License version 2 // ("GPL") and the Apache License version 2 ("ASL"). For the MPL, please see Copyright (c) 2007-2020 VMware, Inc. or its affiliates. All Rights Reserved Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at distributed under the License is distributed on an "AS IS" BASIS, * Based on org.stringtree.json.JSONWriter, licensed under APL and'
Found license 'General Public License 2.0' in '// Copyright (c) 2007-2020 VMware, Inc. or its affiliates. All rights reserved. // This software, the RabbitMQ Java client library, is triple-licensed under the // Mozilla Public License 1.1 ("MPL"), the GNU General Public License version 2 // ("GPL") and the Apache License version 2 ("ASL"). For the MPL, please see Copyright (c) 2007-2020 VMware, Inc. or its affiliates. All Rights Reserved Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at distributed under the License is distributed on an "AS IS" BASIS, * Based on org.stringtree.json.JSONReader, licensed under APL and'

1.312 aws-java-sdk-kms 1.11.415

1.312.1 Available under license:
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2013-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 *
 * Licensed under the Apache License, Version 2.0 (the "License"). You may not use this file except in compliance with
 * the License. A copy of the License is located at
 *
 * http://aws.amazon.com/apache2.0
 *
 * or in the "license" file accompanying this file. This file is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR
 * CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing
 * permissions
 * and limitations under the License.
Found in path(s):
* /opt/cola/permits/1136114712_1613679830.62/0/aws-java-sdk-kms-1-11-415-sources-1.jar/com/amazonaws/services/kms/model/ListKeyPoliciesResult.java
* /opt/cola/permits/1136114712_1613679830.62/0/aws-java-sdk-kms-1-11-415-sources-1.jar/com/amazonaws/services/kms/model/transform/RevokeGrantRequestMarshaller.java
* /opt/cola/permits/1136114712_1613679830.62/0/aws-java-sdk-kms-1-11-415-sources-1.jar/com/amazonaws/services/kms/model/KeyManagerType.java
* /opt/cola/permits/1136114712_1613679830.62/0/aws-java-sdk-kms-1-11-415-sources-1.jar/com/amazonaws/services/kms/model/transform/GetKeyRotationStatusRequestMarshaller.java
* /opt/cola/permits/1136114712_1613679830.62/0/aws-java-sdk-kms-1-11-415-sources-1.jar/com/amazonaws/services/kms/model/InvalidGrantTokenException.java
* /opt/cola/permits/1136114712_1613679830.62/0/aws-java-sdk-kms-1-11-415-sources-1.jar/com/amazonaws/services/kms/model/KeyMetadata.java
* /opt/cola/permits/1136114712_1613679830.62/0/aws-java-sdk-kms-1-11-415-sources-1.jar/com/amazonaws/services/kms/model/ListKeyPoliciesRequestMarshaller.java
* /opt/cola/permits/1136114712_1613679830.62/0/aws-java-sdk-kms-1-11-415-sources-1.jar/com/amazonaws/services/kms/model/GrantOperation.java
* /opt/cola/permits/1136114712_1613679830.62/0/aws-java-sdk-kms-1-11-415-sources-1.jar/com/amazonaws/services/kms/model/CreateGrantResult.java
* /opt/cola/permits/1136114712_1613679830.62/0/aws-java-sdk-kms-1-11-415-sources-1.jar/com/amazonaws/services/kms/model/RevokeGrantRequest.java
* /opt/cola/permits/1136114712_1613679830.62/0/aws-java-sdk-kms-1-11-415-sources-1.jar/com/amazonaws/services/kms/model/CreateKeyResult.java
* /opt/cola/permits/1136114712_1613679830.62/0/aws-java-sdk-kms-1-11-415-sources-1.jar/com/amazonaws/services/kms/model/CreateAliasRequest.java
* /opt/cola/permits/1136114712_1613679830.62/0/aws-java-sdk-kms-1-11-415-sources-1.jar/com/amazonaws/services/kms/model/KeyUnavailableException.java
* /opt/cola/permits/1136114712_1613679830.62/0/aws-java-sdk-kms-1-11-415-sources-1.jar/com/amazonaws/services/kms/model/InvalidKeyUsageException.java
* /opt/cola/permits/1136114712_1613679830.62/0/aws-java-sdk-kms-1-11-415-sources-1.jar/com/amazonaws/services/kms/model/GrantListEntryJsonUnmarshaller.java
* /opt/cola/permits/1136114712_1613679830.62/0/aws-java-sdk-kms-1-11-415-sources-1.jar/com/amazonaws/services/kms/model/ImportKeyMaterialResult.java
* /opt/cola/permits/1136114712_1613679830.62/0/aws-java-sdk-kms-1-11-415-sources-1.jar/com/amazonaws/services/kms/model/KeyIdListEntry.java
* /opt/cola/permits/1136114712_1613679830.62/0/aws-java-sdk-kms-1-11-415-sources-1.jar/com/amazonaws/services/kms/model/GenerateDataKeyResultJsonUnmarshaller.java
* /opt/cola/permits/1136114712_1613679830.62/0/aws-java-sdk-kms-1-11-415-sources-1.jar/com/amazonaws/services/kms/model/EnableKeyResult.java
* /opt/cola/permits/1136114712_1613679830.62/0/aws-java-sdk-kms-1-11-415-sources-1.jar/com/amazonaws/services/kms/model/AlreadyExistsException.java
* /opt/cola/permits/1136114712_1613679830.62/0/aws-java-sdk-kms-1-11-415-sources-1.jar/com/amazonaws/services/kms/model/IncorrectKeyMaterialException.java
* /opt/cola/permits/1136114712_1613679830.62/0/aws-java-sdk-kms-1-11-415-sources-1.jar/com/amazonaws/services/kms/model/GrantListEntryMarshaller.java
* /opt/cola/permits/1136114712_1613679830.62/0/aws-java-sdk-kms-1-11-415-sources-1.jar/com/amazonaws/services/kms/model/CreateKeyResultJsonUnmarshaller.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2011-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 *
 * Licensed under the Apache License, Version 2.0 (the "License"). You may not use this file except in compliance with
 * the License. A copy of the License is located at
 *
 * http://aws.amazon.com/apache2.0
 */
or in the "license" file accompanying this file. This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

*/

Found in path(s):

1.313 hibernate-commons-annotations

5.1.2.Final

1.313.1 Available under license:

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts
as the successor of the GNU Library Public License, version 2, hence
the version number 2.1.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get
it if you want it; that you can change the software and use pieces of
it in new free programs; and that you are informed that you can do
these things.

To protect your rights, we need to make restrictions that forbid
distributors to deny you these rights or to ask you to surrender these
rights. These restrictions translate to certain responsibilities for
you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis
or for a fee, you must give the recipients all the rights that we gave
you. You must make sure that they, too, receive or can get the source
code. If you link other code with the library, you must provide
complete object files to the recipients, so that they can relink them
with the library after making changes to the library and recompiling
it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the
library, and (2) we offer you this license, which gives you legal
permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that
there is no warranty for the free library. Also, if the library is
modified by someone else and passed on, the recipients should know
that what they have is not the original version, so that the original
author's reputation will not be affected by problems that might be
introduced by others.

Finally, software patents pose a constant threat to the existence of
any free program. We wish to make sure that a company cannot
effectively restrict the users of a free program by obtaining a
restrictive license from a patent holder. Therefore, we insist that
any patent license obtained for a version of the library must be
consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the
ordinary GNU General Public License. This license, the GNU Lesser
General Public License, applies to certain designated libraries, and
is quite different from the ordinary General Public License. We use
this license for certain libraries in order to permit linking those
libraries into non-free programs.

When a program is linked with a library, whether statically or using
a shared library, the combination of the two is legally speaking a
combined work, a derivative of the original library. The ordinary
General Public License therefore permits such linking only if the
entire combination fits its criteria of freedom. The Lesser General
Public License permits more lax criteria for linking other code with
the library.

We call this license the "Lesser" General Public License because it does less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.
The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification").

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) The modified work must itself be a software library.

   b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

   c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.
d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.
This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.
6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the
Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

   a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

   b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the
Library), the recipient automatically receives a license from the
original licensor to copy, distribute, link with or modify the Library
subject to these terms and conditions. You may not impose any further
restrictions on the recipients’ exercise of the rights granted herein.
You are not responsible for enforcing compliance by third parties with
this License.

11. If, as a consequence of a court judgment or allegation of patent
infringement or for any other reason (not limited to patent issues),
conditions are imposed on you (whether by court order, agreement or
otherwise) that contradict the conditions of this License, they do not
excuse you from the conditions of this License. If you cannot
distribute so as to satisfy simultaneously your obligations under this
License and any other pertinent obligations, then as a consequence you
may not distribute the Library at all. For example, if a patent
license would not permit royalty-free redistribution of the Library by
all those who receive copies directly or indirectly through you, then
the only way you could satisfy both it and this License would be to
refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any
particular circumstance, the balance of the section is intended to apply,
and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any
patents or other property right claims or to contest validity of any
such claims; this section has the sole purpose of protecting the
integrity of the free software distribution system which is
implemented by public license practices. Many people have made
generous contributions to the wide range of software distributed
through that system in reliance on consistent application of that
system; it is up to the author/donor to decide if he or she is willing
to distribute software through any other system and a licensee cannot
impose that choice.

This section is intended to make thoroughly clear what is believed to
be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in
certain countries either by patents or by copyrighted interfaces, the
original copyright holder who places the Library under this License may add
an explicit geographical distribution limitation excluding those countries,
so that distribution is permitted only in or among countries not thus
excluded. In such case, this License incorporates the limitation as if
written in the body of this License.

13. The Free Software Foundation may publish revised and/or new
versions of the Lesser General Public License from time to time.
Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest
possible use to the public, we recommend making it free software that
everyone can redistribute and change. You can do so by permitting
redistribution under these terms (or, alternatively, under the terms of the
ordinary General Public License).

To apply these terms, attach the following notices to the library. It is
safest to attach them to the start of each source file to most effectively
convey the exclusion of warranty; and each file should have at least the
"copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year>  <name of author>

This library is free software; you can redistribute it and/or
modify it under the terms of the GNU Lesser General Public
License as published by the Free Software Foundation; either
version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public
License along with this library; if not, write to the Free Software
Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the library, if
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the
library 'Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!

1.314 spring-cloud-starter-bootstrap 3.0.4
1.314.1 Available under license :

No license file was found, but licenses were detected in source scan.

<name>Apache License, Version 2.0</name>
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at
distributed under the License is distributed on an "AS IS" BASIS,

Found in path(s):
* /opt/cola/permits/1223884317_1635968840.91/0/spring-cloud-starter-bootstrap-3-0-4-sources-jar/META-INF/maven/org.springframework.cloud/spring-cloud-starter-bootstrap/pom.xml
No license file was found, but licenses were detected in source scan.

/*
 *
 * Copyright 2013-2020 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 *
 */

Found in path(s):
* /opt/cola/permits/1223884317_1635968840.91/0/spring-cloud-starter-bootstrap-3-0-4-sources-jar/org/springframework/cloud/bootstrap/marker/Marker.java

1.315 spring-framework 5.3.12
1.315.1 Available under license :
No license file was found, but licenses were detected in source scan.

/*
 *
 * Copyright 2002-2010 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-
jar/org/springframework/core/convert/ConversionException.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2002-2009 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-
jar/org/springframework/core/type/classreading/MetadataReader.java

No license file was found, but licenses were detected in source scan.

// All rights reserved.
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions
// are met:
// 1. Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
// 2. Redistributions in binary form must reproduce the above copyright
// notice, this list of conditions and the following disclaimer in the
// documentation and/or other materials provided with the distribution.
// 3. Neither the name of the copyright holders nor the names of its
// this software without specific prior written permission.

Found in path(s):
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-
jar/org/springframework/asm/RecordComponentVisitor.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-
jar/org/springframework/asm/TypeReference.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-
jar/org/springframework/asm/FieldWriter.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-
jar/org/springframework/asm/AnnotationVisitor.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-
jar/org/springframework/asm/RecordComponentWriter.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-
jar/org/springframework/asm/ModuleWriter.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-
jar/org/springframework/asm/Context.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-
jar/org/springframework/asm/SymbolTable.java

No license file was found, but licenses were detected in source scan.

/
* Copyright 2002-2018 the original author or authors.
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-
jar/org/springframework/core/NamedInheritableThreadLocal.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-
jar/org/springframework/core/ParameterizedTypeReference.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-
jar/org/springframework/core/env/ReadOnlySystemAttributesMap.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-
jar/org/springframework/core/io/support/LocalizedResourceHelper.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-
jar/org/springframework/core/io/VfsResource.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-
jar/org/springframework/core/io/OverridingClassLoader.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-
jar/org/springframework/util/ResourceUtils.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-
jar/org/springframework/util/CompositeIterator.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-
jar/org/springframework/core/io/Resource.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/util/comparator/NullSafeComparator.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/xml/ListBasedXMLEventReader.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/core/NamedThreadLocal.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/lang/UsesJava8.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/core/convert/support/ConversionServiceFactory.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/core/task/support/ExecutorServiceAdapter.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/core/MethodIntrospector.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/core/convert/converter/ConverterFactory.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/util/xml/AbstractXMLStreamReader.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/util/InstanceFilter.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/util/ConcurrentThrottleSupport.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/util/Concurrent/ComplettableToListenableFutureAdapter.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/core/env/ConfigurableEnvironment.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/util/concurrent/SettableListenableFuture.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/util/concurrent/CompletableToListenableFutureAdapter.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/core/env/ConfigurableEnvironment.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/util/Concurrent/ListenbleFuture.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/core/env/ConfigurableEnvironment.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/util/concurrent/SettableListenableFuture.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/util/concurrent/ListenbleFuture.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/core/env/ConfigurableEnvironment.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/util/concurrent/ListenbleFutureCallback.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/core/ExceptionDepthComparator.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-jar/org/springframework/core/env/Environment.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-jar/org/springframework/core/env/PropertySources.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-jar/org/springframework/core/io/support/ResourcePropertySource.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-jar/org/springframework/core/ParameterNameDiscoverer.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-jar/org/springframework/core/io/InputStreamResource.java

No license file was found, but licenses were detected in source scan.

# Copyright 2002-2020 the original author or authors.
#
# Licensed under the Apache License, Version 2.0 (the "License");
# you may not use this file except in compliance with the License.
# You may obtain a copy of the License at
#
#      https://www.apache.org/licenses/LICENSE-2.0
#
# Unless required by applicable law or agreed to in writing, software
# distributed under the License is distributed on an "AS IS" BASIS,
# WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
# See the License for the specific language governing permissions and
# limitations under the License.

org.springframework.core.ReactiveAdapterRegistry$SpringCoreBlockHoundIntegration

Found in path(s):
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-jar/META-INF/services/reactor.blockhound.integration.BlockHoundIntegration

No license file was found, but licenses were detected in source scan.
* Copyright 2002-2015 the original author or authors.
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
*   https://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-
jar/org/springframework/core/serializer/support/DeserializingConverter.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-
jar/org/springframework/core/serializer/DefaultSerializer.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-
jar/org/springframework/core/env/SystemEnvironmentPropertySource.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-
jar/org/springframework/core/convert/support/StringToCurrencyConverter.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-
jar/org/springframework/core/convert/support/StringToCharsetConverter.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-
jar/org/springframework/core/convert/converter/ConditionalGenericConverter.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-
jar/org/springframework/core/convert/support/FallbackObjectToStringConverter.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-
jar/org/springframework/core/convert/support/SerializationDelegate.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-
jar/org/springframework/util/ResizableByteArrayOutputStream.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-
jar/org/springframework/core/convert/support/StringToCharsetConverter.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-
jar/org/springframework/util/AlternativeJdkIdGenerator.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2007 the original author or authors.
 */
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

*/

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2017 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/core/type/filter/RegexPatternTypeFilter.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/core/io/ContextResource.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/core/style/ValueStyler.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/lang/NonNullApi.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/util/xml/StaxResult.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/core/convert/support/AbstractConditionalEnumConverter.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/core/convert/converter/GenericConverter.java

Open Source Used In security_automation identity-guest-cs 2899
No license file was found, but licenses were detected in source scan.
/*
 * Copyright 2002-2019 the original author or authors.
 * *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * *
 *   https://www.apache.org/licenses/LICENSE-2.0
 * *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/core/annotation/AnnotationAwareOrderComparator.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/core/io/buffer/DataBuffer.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/util/xml/StaxUtils.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/core/log/CompositeLog.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/core/codec/DataBufferEncoder.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/core/annotation/MergedAnnotationPredicates.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/util/MultiValueMap.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/core/io/FileSystemResource.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/core/codec/ByteBufferEncoder.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/core/io/charset/CharsetEncoder.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/core/io/charset/CharsetDecoder.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/core/io/DefaultResourceLoader.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/core/io/ResourceLoader.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/core/io/ResourceLoaderSupport.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/core/io/ResourcePatternResolver.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/core/io/ResourcePatternRegistry.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/core/io/ResourcePatternSet.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/core/io/ResourcePatternSetImpl.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/core/io/ResourcePatternResolverFacade.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/core/io/ResourcePatternMatcher.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/core/io/ResourcePatternMatcherImpl.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/core/io/ResourcePatternRegistryImpl.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/core/io/ResourcePatternSetImpl.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/core/io/ResourcePatternSetRegistryImpl.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/core/io/ResourcePatternMatcherFacade.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/core/io/ResourcePatternMatcherFacadeImpl.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/core/io/ResourcePatternResolverFacadeImpl.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/core/io/ResourcePatternResolverFacadeImpl.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/core/io/ResourcePatternResolverFacadeImpl.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/core/io/ResourcePatternResolverFacadeImpl.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/core/io/ResourcePatternResolverFacadeImpl.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/core/io/ResourcePatternResolverFacadeImpl.java
Open Source Used In security_automation identity-guest-cs 2903
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2016 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *      https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-jar/org/springframework/core/env/CommandLineArgs.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-jar/org/springframework/core/configurableObjectInputstream.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-jar/org/springframework/core/convert/support/StringToPropertiesConverter.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-jar/org/springframework/core/convert/support/EnumToIntegerConverter.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-jar/org/springframework/core/convert/support/ConverterRegistry.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-jar/org/springframework/core/convert/support/BackOffExecution.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/core/type/filter/AssignableTypeFilter.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/util/concurrent/FailureCallback.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/core/env/ConfigurablePropertyResolver.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/core/convert/support/CollectionToStringConverter.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/core/io/ProtocolResolver.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/core/convert/support/ZonedDateTimeToCalendarConverter.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/core/convert/support/ZoneIdToTimeZoneConverter.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/core/convert/support/StringToTimeZoneConverter.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/core/convert/support/CharacterToNumberFactory.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/core/convert/support/EnumToStringConverter.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/objenesis/SpringObjenesis.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/util/backoff/BackOff.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2021 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *    https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-
jar/org/springframework/core/io/buffer/DefaultDataBuffer.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-
jar/org/springframework/util/ConcurrentReferenceHashMap.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-
jar/org/springframework/core/io/AbstractFileResolvingResource.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-
jar/org/springframework/core/io/InputStreamSource.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-
jar/org/springframework/core/Constants.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-
jar/org/springframework/core/CoroutinesUtils.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-
jar/org/springframework/core/metrics/jfr/FlightRecorderStartupStep.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2012-2020 the original author or authors.
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
*   https://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-
jar/org/springframework/core/metrics/jfr/FlightRecorderStartupStep.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2002-2011 the original author or authors.
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
*   https://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-jar/org/springframework/core/convert/support/ConfigurableConversionService.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2002,2003,2004 The Apache Software Foundation
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* *
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-jar/org/springframework/cglib/proxy/Enhancer.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2002-2012 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* *
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
* */
Found in path(s):
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-jar/org/springframework/core/task/TaskRejectedException.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-jar/org/springframework/core/task/support/ConcurrentExecutorAdapter.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-jar/org/springframework/core/task/TaskTimeoutException.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-jar/org/springframework/cglib/SpringCglibInfo.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2014 the original author or authors.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * https://www.apache.org/licenses/LICENSE-2.0
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-jar/org/springframework/cglib/SpringCglibInfo.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2020 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *    https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-jar/org/springframework/core/convert/support/StringToCollectionConverter.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-jar/org/springframework/core/type/filter/AbstractClassTestingTypeFilter.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-jar/org/springframework/util/ExceptionTypeFilter.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-jar/org/springframework/core/convert/support/ArrayToStringConverter.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-jar/org/springframework/core/convert/support/ArrayToObjectConverter.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-jar/org/springframework/core/convert/support/PropertiesToStringConverter.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-jar/org/springframework/cglib/core/SpringNamingPolicy.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-jar/org/springframework/util/concurrent/MonoToListenableFutureAdapter.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-jar/org/springframework/util/xml/StaxStreamHandler.java
Open Source Used In security_automation identity-guest-cs 2015
jar/org/springframework/core/PrioritizedParameterNameDiscoverer.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-
jar/org/springframework/core/task/SimpleAsyncTaskExecutor.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-
jar/org/springframework/core/io/support/ResourceArrayPropertyEditor.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-
jar/org/springframework/core/io/support/PropertiesLoaderUtils.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-
jar/org/springframework/core/annotation/MergedAnnotationsCollection.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-
jar/org/springframework/core/annotation/MergedAnnotationCollectors.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-
jar/org/springframework/util/Assert.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-
jar/org/springframework/util/LinkedMultiValueMap.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-
jar/org/springframework/util/DefaultPropertiesPersister.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-
jar/org/springframework/core/convert/support/StringToBooleanConverter.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-
jar/org/springframework/core/convert/support/StringToNumberConverterFactory.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-
jar/org/springframework/core/io/UrlResource.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-
jar/org/springframework/core/convert/converter/ConvertingComparator.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-
jar/org/springframework/core/ReactiveTypeDescriptor.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-
jar/org/springframework/core/convert/support/ConvertingComparator.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-
jar/org/springframework/core/ReactiveTypeDescriptor.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-
jar/org/springframework/util/CustomizableThreadCreator.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources-
jar/org/springframework/core/log/LogAccessor.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/core/convert/support/StringToUUIDConverter.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/core/env/MutablePropertySources.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/core/env/EnumerablePropertySource.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/core/Conventions.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2003,2004 The Apache Software Foundation
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
*     https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/cglib/proxy/MethodProxy.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/cglib/core/AbstractClassGenerator.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/cglib/core/ReflectUtils.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/cglib/core/AsmApi.java
* /opt/cola/permits/1220178500_1635319458.1/0/spring-core-5-3-12-sources.jar/org/springframework/cglib/core/KeyFactory.java

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: www.cisco.com/go/trademarks. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

©2022 Cisco Systems, Inc. All rights reserved.