Math

The `Math` action element is used to evaluate basic mathematical expressions. The mathematical expression is composed of operators and functions in the form of a string which is passed as a setting to the element, parsed and evaluated at runtime. The result is a double value stored as a string in either element data or session data. All common arithmetic operators are supported. Boolean operators are also fully supported. Boolean expressions are evaluated to be either 1.0 or 0.0 (*true* or *false* respectively).

Examples

Expression: 2 * 4 Result: 8.0	Expression: sqrt(16) Result: 4.0	Expression: {Data.Session.myNumber} == 4 Result: 1.0

Settings

Name (Label)	Type	Req'd	Single Setting Value	Substitution Allowed	Default	Notes
Type (Type)	string enum	Yes	`true`	`false`	`Element`	This setting specifies the type of data that will store the result of the mathematical expression. Possible values are: `Element` | `Session`. Default = `Element`.

Name (Name)	string	Yes	`true`	`true`	*None*	This setting specifies the name to assign to the data that will store the result of the mathematical expression.
Expression (Expression)	string	Yes	`true`	`true`	*None*	This setting specifies the mathematical expression to parse and evaluate. For supported operators and functions see tables below.

Operators and Functions

Operator Name	Operator	Function Name	Syntax
Power	^	Sine	sin(x)
Boolean Not	!	Cosine	cos(x)
Unary Plus, Unary Minus	+x, -x	Tangent	tan(x)
Modulus	%	Arc Sine	asin(x)
Division	/	Arc Cosine	acos(x)
Multiplication	*	Arc Tangent	atan(x)
Addition, Subtraction	+, -	Arc Tangent (with 2 parameters)	atan2(y, x)
Less or Equal, More or Equal	<=, >=	Hyperbolic Sine	sinh(x)
Less Than, Greater Than	<, >	Hyperbolic Cosine	cosh(x)
Not Equal, Equal	!=, ==	Hyperbolic Tangent	tanh(x)
Boolean And	&&	Inverse Hyperbolic Sine	asinh(x)
Boolean Or	||	Inverse Hyperbolic Cosine	acosh(x)
		Inverse Hyperbolic Tangent	atanh(x)
		Natural Logarithm	ln(x)
		Logarithm base 10	log(x)
		Exponential	exp(x)
		Absolute Value / Magnitude	abs()
		Modulus	mod()
		Square Root	sqrt()
		Sum	sum()
		If	if()

Element Data

Element data is created *only* when the `type` setting is set to *Element*. In all other cases, no element data is created.

Name	Type	Notes
[value of setting "name"]	string	The result of the mathematical expression.

Session Data

Session data is created *only* when the `type` setting is set to *Session*. In all other cases, no session data is created.

Name	Type	Notes
[value of setting "name"]	string	The result of the mathematical expression.

Exit States

Name	Notes
done	The mathematical expression was evaluated and the result was stored as either element data or session data.

Folder and Class Information

Studio Element Folder Name	Class Name
Math	com.audium.server.action.math.MathAction

Events

Name (Label)	Notes
Event Type	You can select Java Exception as event handler type.

The output of the Customer_Lookup element can be in JSON format . To know more about parsing the JSON Data refer to "Parsing JSON Data" section in *User Guide for Cisco Unified CVP VXML Server and Cisco Unified Call Studio.*