Procedural Tasks

This chapter contains the following sections:

Acquire lock on a named Resource

Summary

Acquire a lock on a named resource.

Description

This task acquires a lock on any named resource. If the named resource is free, meaning no other task holds a lock on it, then the acquisition of the resource succeeds and the task completes successfully. If another task holds a lock on the named resource, then the calling task is suspended until the other task releases the lock. Re-entrant acquisition is permitted and is always successful. A lock can be acquired by a sub-task within a compound task. If the lock is already held by the compound task (or its parent workflow, or a parent's parent), the acquisition is considered re-entrant and always succeeds. For example: Workflow WF-A acquires a lock L1 and spawns a Compound Task CT1. If a sub-task in CT1 tries to acquire L1, then the acquisition succeeds. Similarly, suppose workflow WF-B acquires a lock L2 and spawns CT2, which in turn spawns CT3. If a sub-task in CT3 tries to acquire L2, it will succeed because WF-B is an (indirect) parent of CT2. A workflow should relinquish the lock on the resource (using the Release Resource Lock task) as soon as it is done using the resource. Behavior during Rollback of an SR When a workflow is rolled back, the tasks are executed in reverse order of normal execution. When a workflow containing an AcquireLock followed by a ReleaseLock task is subjected to Rollback, the roles of AcquireLock and ReleaseLock are reversed. For example, consider this workflow: AcquireLock(R1) -> Task1 -> Task2 -> ReleaseLock(R1) In this example, AcquireLock(R1) acquires lock on resource R1 and ReleaseLock(R1) relinquishes the lock on R1. As part of a rollback, the ReleaseLock(R1) task is executed first. But since the intent is ensure safe access to resouce R1, this ReleaseLock behaves like an AcquireLock(R1) task and acquires the lock on R1. Subsequently, when the AcquireLock(R1) task is invoked it executes the ReleaseLock(R1) operation, relinquishing the lock on resource R1.

Inputs

Input	Description	Mappable To Type	Mandatory
Resource name	Name of resource that must be locked	gen_text_input	Y

Outputs

Output	Description	Type
ResourceName	Name of the Resource that was acquired for exclusive access	gen_text_input

Clone Workflow

Summary
Description
Inputs

Input	Description	Mappable To Type	Mandatory
Workflow	Workflow to clone	workflowSelector	Y
Cloned Workflow Name	Workflow Version to clone	gen_text_input	Y
New Folder	Select to give a new Folder name	Boolean	
Folder Name	Folder Name	gen_text_input	
New Folder Name	Folder Name	gen_text_input	Y

Outputs

Output	Description	Type
OUTPUT_SOURCE_WF_NAME	Source WF Name	gen_text_input
OUTPUT_SOURCE_WF_ID	Source WF ID	gen_text_input
OUTPUT_CLONE_WF_NAME	Source WF Name	gen_text_input
OUTPUT_CLONE_WF_ID	Source WF Name	gen_text_input

Conditional Task

Summary

Select the next Task based on the evaluation of Boolean statements, similar to a "switch" statement in most programming languages.

Description

This task takes as input two or more conditions and associated labels. The labels are mapped to other tasks. At runtime, the conditions are evaluated in order and control is passed to the task associated with the first condition that evaluates to True. If no condition evaluates to True, the Conditional Task fails.

Inputs

Input	Description	Mappable To Type	Mandatory
List of Conditions	Add List of Conditions		

Outputs

No Outputs

End Loop

Summary

End a loop.

Description

This task marks the end of a loop in the workflow.

Inputs

Input	Description	Mappable To Type	Mandatory
No Inputs			

Outputs

No Outputs

If Else

Summary

If Else Conditional Task.

Description

This task evaluates the condition given as input. Depending on the result, it passes control to a different task.

Inputs

Input	Description	Mappable To Type	Mandatory
Specify the condition	Specify the condition		Y

Outputs

No Outputs

Release lock on a named Resource

Summary

Release a lock held on a named resource.

Description

This task releases a lock held on a named resource. Lock acquisition is described in the Acquire Resource Lock task. Acquired locks must be released when the acquiring task is done with the resource. This implies that, within a workflow, the number of Acquire operations should equal the number of Release operations. A spurious Release (release of a lock that was never held) is ignored. A workflow should only release a lock that it successfully acquired using the Acquire Resource Lock task. For behavior of locks during rollback, please see AcquireLock task

Inputs

Input	Description	Mappable To Type	Mandatory
Resource name	Name of resource that must be unlocked	gen_text_input	Y

Outputs

Output	Description	Type
ResourceName	Name of the Resource that was released from exclusive access	gen_text_input

Resume Task

Summary

Wait until a date and time.

Description

This task pauses the workflow execution until the date and time specified in the input.

Inputs

Input	Description	Mappable To Type	Mandatory
Time	Time to execute	date_time	Y

Outputs

No Outputs

Start Loop

Summary

Start a loop.

Description

This task begins a loop in the workflow based on the input condition.

Inputs

Input	Description	Mappable To Type	Mandatory
List based iteration	List based iteration		
Input for list based iteration	Input for list based iteration	gen_text_input	Y
User Input to assign iterated values	User Input to assign iterated values		
Count based iteration	Count based iteration		
Number of times to loop	Number of times to loop	gen_text_input	Y

Outputs

Output	Description	Type
START_LOOP_OUTPUT	Each Iteration Value	gen_text_input