

Module 5: Workflow, Citrix and Data Manipulation

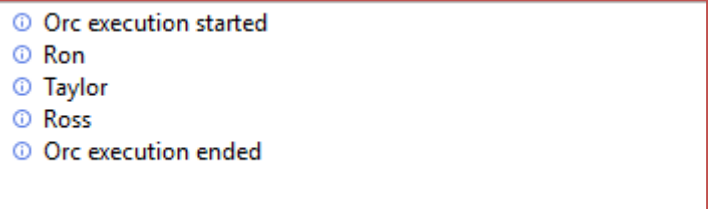
Demo 2 – Solution

edureka!

edureka!

Demo 2

- Automate a process of storing a name in a list
 1. Create a list of type String
 2. Add some default names to it
 3. Use invoke method to add a new name to the list
 4. Display the final list in the output screen



```
❶ Orc execution started
❶ Ron
❶ Taylor
❶ Ross
❶ Orc execution ended
```

Demo 2 – Solution

- **Step 1:** Create a blank project and, on the Design tab, in the File group, select New -> Sequence. The New Sequence Diagram window is displayed. Give a name to the Sequence.
- **Step 2:** Create 2 variables, one of type String[] and the other of type List<String>. We have created anArray of type String[] and aList of type List<String>.
- **Step 3:** In the default value of String[] type variable, set two entries in the format shown below in the screenshot. In the default value of List<String> type variables, set the value as shown below:
new List(of String)from{"Ron", "Taylor"}

Name	Variable type	Scope	Default
anArray	String[]	array	{"Ron", "Taylor"}
aList	List<String>	array	new List(of String)from{"Ron", "Taylor"}

- **Step 4:** Add an **Invoke Method** Activity and set **aList** variable in the Target Object field and set Method Name as **Add**.

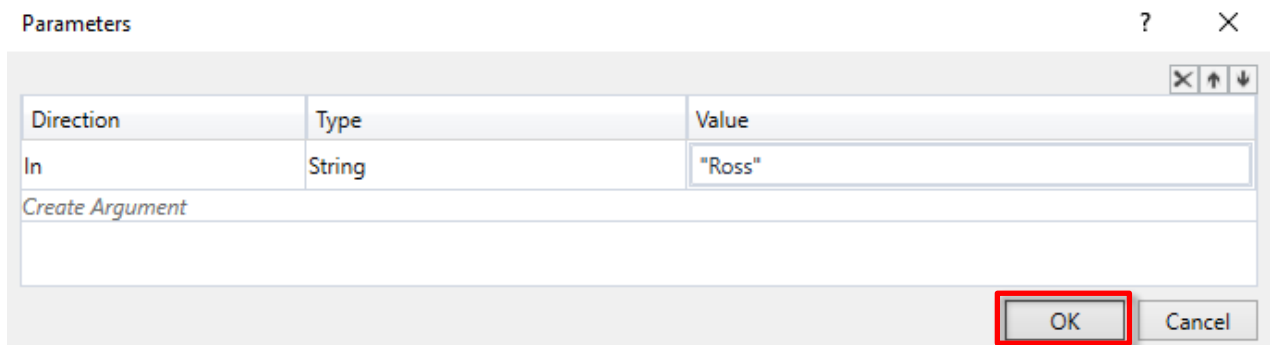
Invoke method

TargetType: (null)

TargetObject: aList

MethodName: Add

- **Step 5:** In the Invoke Method Parameters Property, create an Argument and enter a new name in the Value as shown below and click OK:



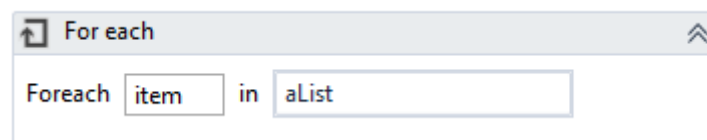
The Parameters dialog box is shown with a table containing one row. The 'Direction' column has the value 'In', the 'Type' column has the value 'String', and the 'Value' column has the value '"Ross"'. Below the table is a 'Create Argument' button. At the bottom right, the 'OK' button is highlighted with a red rectangle.

Direction	Type	Value
In	String	"Ross"

Create Argument

OK Cancel

- **Step 6:** Add a **For each** loop activity under Invoke Method. Set **Foreach** item in **aList**

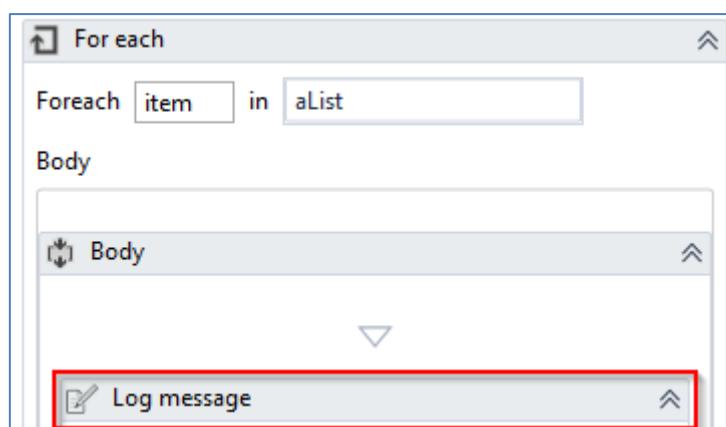


The 'For each' loop configuration is shown. The 'Foreach' dropdown is set to 'item', and the 'in' dropdown is set to 'aList'.

For each

Foreach item in aList

- **Step 7:** In the **For each** loop's Body, add a **Log Message** Activity.



The 'For each' loop body configuration is shown. The 'Body' section contains a 'Log message' activity, which is highlighted with a red rectangle.

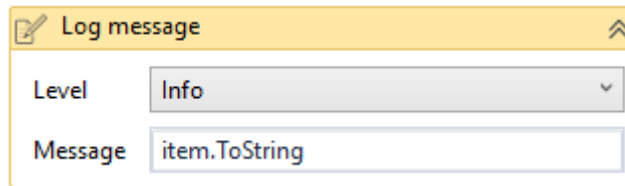
For each

Foreach item in aList

Body

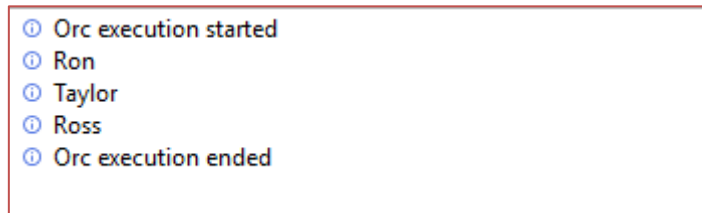
Log message

- **Step 8:** Set the **Level** field to **Info** and **Message** field to **item.ToString**



The screenshot shows a 'Log message' dialog box with a yellow header. It contains two fields: 'Level' with a dropdown menu currently showing 'Info', and 'Message' with a text input field containing the expression 'item.ToString'.

- **Step 9:** **Run** the Sequence. You will see the name which you have entered in Step 5, is added in the collection.



The screenshot displays a list of items within a collection, each preceded by a blue circular icon with a white dot. The items are: 'Orc execution started', 'Ron', 'Taylor', 'Ross', and 'Orc execution ended'.

edureka!