**How to Use Methods in SoapUI for Bulk Test Execution**

### What is object oriented programming?

[Object oriented programming](http://en.wikipedia.org/wiki/Object-oriented_programming) is a programming language specification that deals with “objects”. Using this concept, complex desktop/web/mobile and business applications are often developed. A brief introduction to the components of object oriented programming concept is below:

* **Classes and Objects**
  + Consists of methods and properties. Methods are set of instructions that perform specific action. It helps us to reuse the code that is already defined. Properties are the variables to store the values temporarily.
* **Inheritance**
  + This is very useful component in OOP. It helps us create a new class from an existing class so the properties and methods can be reused or accessed in the derived class directly. Technically, the main class is called as Base / super class and the inherited class is known as derived class.
* **Encapsulation**
  + Using this concept, we can hide the class information or protect properties of the class by accessing from outside.
* **Polymorphism**
  + **Poly** refers “many” and **morphism** means “forms” that means an object can be defined in many forms.

**What are objects?**

An [object](https://docs.oracle.com/javase/tutorial/java/concepts/object.html) is a container that consists of properties and methods that perform certain action(s).

Objects are considered as real world materials like smart phone, tablet, monitor etc. it has characteristics and activities.

Let’s take human being as an example for object. As a person, we have different characters and behavior among us like face, skin color, living style etc. we are doing regular activities such breathing, speaking, walking and so forth. So an object can include characteristics and actions implicitly.

In the object oriented programming paradigm, an object is an instance of a class which consists of properties and methods also it is represented as blue print of the class. A class is generally defined as following format.

class <class name>  
{  
                [access modifiers:]  
                <properties declaration>

                <methods>  
}

As we discussed in our previous tutorials, groovy script supports java code directly without having to import built-in libraries explicitly. Groovy script editor does not allow defining user-defined classes directly but defining methods in groovy is straightforward. Once done these methods can be invoked anywhere in the script. The methods in groovy scripts can be defined as below:

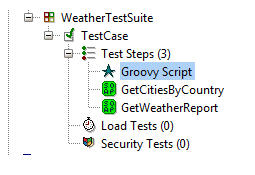
<return type> <method name> [ parameters ]  
{  
                <statements>  
[return]  
}

**Example:**

In the following example, we are going to define a method with some input parameters. Later the parameter values will be assigned to the global properties. Then, we will pass the property values to the service request as an input.

**Here are steps:**

* Create a project with http://www.webservicex.net/globalweather.asmx?WSDL URL
* And then add test suite and test cases as shown in the below screenshot

[](http://cdn.softwaretestinghelp.com/wp-content/qa/uploads/2015/06/Methods-in-SoapUI-1.jpg)

Double click on the groovy script test step and Copy and paste the following script.

|  |  |
| --- | --- |
| 1 | def countryName = "India" |
| 2 | def cityName = "Hyderabad" | |

|  |  |
| --- | --- |
| 3 |  |
| 4 | void CreateAndPassProperties(String countryName, String cityName) | |

|  |  |
| --- | --- |
| 5 | { |
| 6 | // Add Global Properties | |

|  |  |  |
| --- | --- | --- |
| 7 | com.eviware.soapui.SoapUI.globalProperties.addProperty( "CountryName") | |
| 8 | com.eviware.soapui.SoapUI.globalProperties.addProperty( "CityName") |

|  |  |  |
| --- | --- | --- |
| 9 |  | |
| 10 | | // Assign values to the global properties | |

|  |  |  |
| --- | --- | --- |
| 11 | com.eviware.soapui.SoapUI.globalProperties. | |
| 12 | setPropertyValue( "CountryName", countryName ) |

|  |  |  |
| --- | --- | --- |
| 13 | com.eviware.soapui.SoapUI.globalProperties. | |
| 14 | setPropertyValue( "CityName", cityName ) |

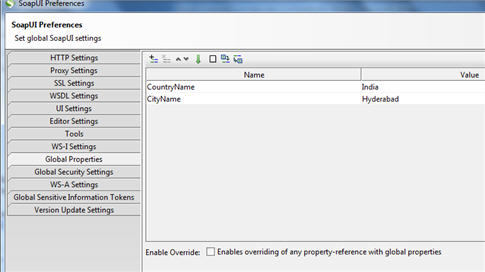
|  |  |  |
| --- | --- | --- |
| 15 | } | |
| 16 |  |

|  |  |
| --- | --- |
| 17 | // Method Invoking |
| 18 | CreateAndPassProperties(countryName, cityName) | |

|  |  |
| --- | --- |
| 19 | log.info("Testcase execution is completed successfully.") |

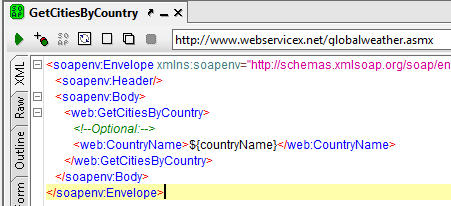
In the above script, we are defining local variables and assigning string data to each variable. Following that, we define a method called **CreateAndPassProperties** with two parameters – **countryName** and **cityName**.

Within the method, the first two lines are used to define the global properties. These can be seen by using the “File->Preferences->Global Properties” as below. **Initially before the test gets executed this will be empty.**

[](http://cdn2.softwaretestinghelp.com/wp-content/qa/uploads/2015/06/Methods-in-SoapUI-2.jpg)

Once done go to the request test steps like **GetCitiesByCountry** and **GetWeatherReport** and make necessary changes in the input request as shown in the below screen shots.

**GetCitiesByCountry Web Service:**

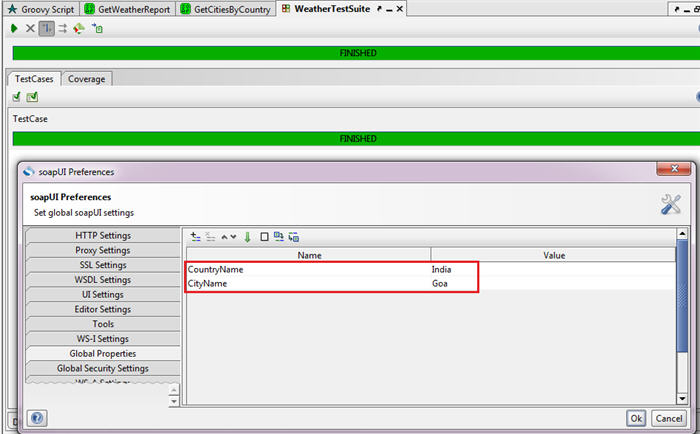
[](http://cdn.softwaretestinghelp.com/wp-content/qa/uploads/2015/06/Methods-in-SoapUI-3.jpg)

**GetWeatherReport Webservice:**

------------

Now we can execute the test suite and verify the results with the following screenshot.

(Click image for enlarged view)

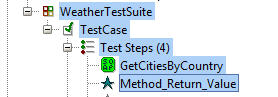
[](http://cdn.softwaretestinghelp.com/wp-content/qa/uploads/2015/06/Methods-in-SoapUI-5.jpg)

This way we can define multiple methods for various purposes in groovy script test step which can help us execute bulk test steps at once. Methods also allow us to pass arrays to handle bulk data.

### Methods and Arrays:

We will now to assign array values to a global property and then pass it to the service. During test suite execution, the script will pass input data to the service and the service will then process the data. Finally it will send the respective response in SoapUI.

Let us create a test suite and test steps as shown in this screenshot.

[](http://cdn2.softwaretestinghelp.com/wp-content/qa/uploads/2015/06/Methods-in-SoapUI-6.jpg)

* Double click on the **GetCitiesByCountry** service and make the change in the request as shown in the following screenshot.

[](http://cdn2.softwaretestinghelp.com/wp-content/qa/uploads/2015/06/Methods-in-SoapUI-7.jpg)

* Double click on the **Method\_Return\_Value** test step and write the following script:

|  |  |
| --- | --- |
| 1 | def MAX\_LIMIT = 5 |
| 2 | def countries = new Object[MAX\_LIMIT] | |

|  |  |
| --- | --- |
| 3 |  |
| 4 | countries[0] = "India" | |

|  |  |
| --- | --- |
| 5 | countries[1] = "US" |
| 6 | countries[2] = "Mauritius" | |

|  |  |
| --- | --- |
| 7 | countries[3] = "Cyprus" |
| 8 | countries[4] = "Austria" | |

|  |  |  |
| --- | --- | --- |
| 9 |  | |
| 10 | | // Invoke Method | |

|  |  |  |
| --- | --- | --- |
| 11 | GetCountries(countries); | |
| 12 |  |

|  |  |
| --- | --- |
| 13 | // Method Definition |
| 14 | void GetCountries(Object[] countries) | |

|  |  |
| --- | --- |
| 15 | { |
| 16 | for(int i=0; i<5; i++) | |

|  |  |
| --- | --- |
| 17 | { |
| 18 | // Assign values to the global properties and call the servive | |

|  |  |  |
| --- | --- | --- |
| 19 | com.eviware.soapui.SoapUI.globalProperties.setPropertyValue | |
| 20 | ("CountryName", countries[i] ) |

|  |  |
| --- | --- |
| 21 |  |
| 22 | // Call GetCitiesByCountry service to run | |

|  |  |  |
| --- | --- | --- |
| 23 | def testStep = testRunner.testCase.testSteps['GetCitiesByCountry']; | |
| 24 | testStep.run(testRunner,context); |

|  |  |  |
| --- | --- | --- |
| 25 | log.info('Method is called ' + i + 'time(s)'); | |
| 26 | } |

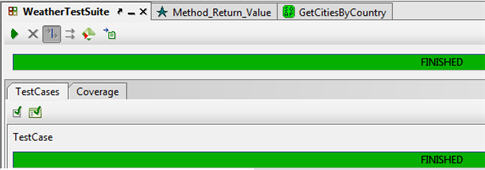
|  |  |
| --- | --- |
| 27 | } |

In the above script, we have defined an array called **countries** which holds 5 country names.

Following that, we are invoking **GetCountries** method by passing **countries** array. Now the execution flow moves to the method definition where we iterate the loop five times. Each time array element that is country name will be assigned to the global property (i.e. CountryName). Then, **GetCitiesByCountry** test step gets called.

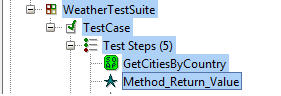
Since we already mentioned global property name in the service request, actual country name will be taken as input and processed accordingly.

Once this test suite is executed successfully, verify the response to determine the service execution status. Let us have a look at the following screenshot.

[](http://cdn.softwaretestinghelp.com/wp-content/qa/uploads/2015/06/Methods-in-SoapUI-8.jpg)

Now you know the purpose of the methods there is another feature that we need to discuss in this topic and that is Methods can be return values to the caller through output values.

To do so, we employ the “**return**” keyword. Let us take **GetCitiesByCountry** service for this example. Add one more groovy script test step under **WeatherTestSuite** as shown in the screenshot.

[](http://cdn2.softwaretestinghelp.com/wp-content/qa/uploads/2015/06/Methods-in-SoapUI-9.jpg)

Double click on the service name and make the changes if needed. Then write the following script in the groovy script editor.

|  |  |
| --- | --- |
| 1 | // Method definition |
| 2 | String ReturnCountryName(String CName) | |

|  |  |
| --- | --- |
| 3 | { |
| 4 | CName = 'Mauritius'; | |

|  |  |  |
| --- | --- | --- |
| 5 | return CName; | |
| 6 | } |

|  |  |
| --- | --- |
| 7 |  |
| 8 | // Invoke method | |

|  |  |  |  |
| --- | --- | --- | --- |
| 9 | String  getMethodValue = ReturnCountryName('US'); | | |
| 10 | |  |

|  |  |
| --- | --- |
| 11 | // Assign value to the global property |
| 12 | com.eviware.soapui.SoapUI.globalProperties.setPropertyValue | |

|  |  |  |
| --- | --- | --- |
| 13 | ( "CountryName", getMethodValue ) | |
| 14 |  |

|  |  |
| --- | --- |
| 15 | // Call GetCitiesByCountry service to run |
| 16 | def testStep = testRunner.testCase.testSteps['GetCitiesByCountry']; | |

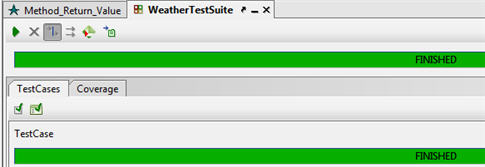
|  |  |  |
| --- | --- | --- |
| 17 | testStep.run(testRunner,context); | |
| 18 |  |

|  |  |
| --- | --- |
| 19 | // Message |
| 20 | log.info('Testsuite executed successfully'); | |

The above is a slight modification of the previous script. In the method definition, we are passing a country name as string type and then we assign new country name to the method variable. So the “US” string will be changed to “Mauritius”after the execution.

This will be sent to the **getMethodValue** variable. Remaining code is same as we discussed earlier. It will assign property value to the global property and then service test step will be called by **testStep.run(…)**method.

To execute the test suite, double click on the test suite name and then click Run icon. The result would be as below:

[](http://cdn2.softwaretestinghelp.com/wp-content/qa/uploads/2015/06/Methods-in-SoapUI-10.jpg)

### Conclusion

**Here is a quick recap of all the concepts we discussed so far:**

* Object oriented programming concept plays a main role in all the major programming languages.
* Objects are the main components that contain properties and methods.
* Objects are the blueprint of a class.
* Without defining class, it is not possible to instantiate an object.
* Methods are one of the parts in the class. These are used to avoid repeated code.

A few samples have been provided in this article to execute test cases. Implement these samples to your real time web services and test suites to make you more comfortable and to become an expert in the script writing.