**Fetching custom properties**

There are two ways to fetch properties



Or



You can give the name of the test case or the test step and then link to their properties.

**Setting Custom Properties**



**Converting JSON response to MAP and saving values from Map to a variable**



Convert the response to a map using Slurper

groovy.json.JsonSlurper slurper = new groovy.json.JsonSlurper();

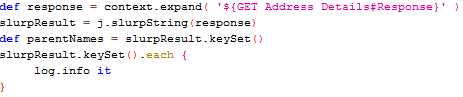
def mappedResult = slurper.parseText (Response)

Fetch values of each parameter like so

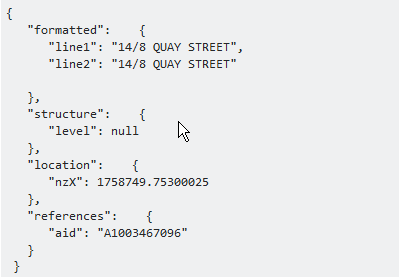
def Actual\_soType=mappedResult[requiredIndex]["soType"]

def Actual\_soStage=mappedResult[requiredIndex]["soStage"]

**Getting all the Key/Parents nodes only**



Output of the above code on



Would be Formatted, structure, location and references.

**Using HashMaps in groovy**

Ways to get the value

Java.Util.Map Temp2=[node:”vishal”, expected:Username]

temp2.get('expected') gives the value of username

temp2[1] represents username

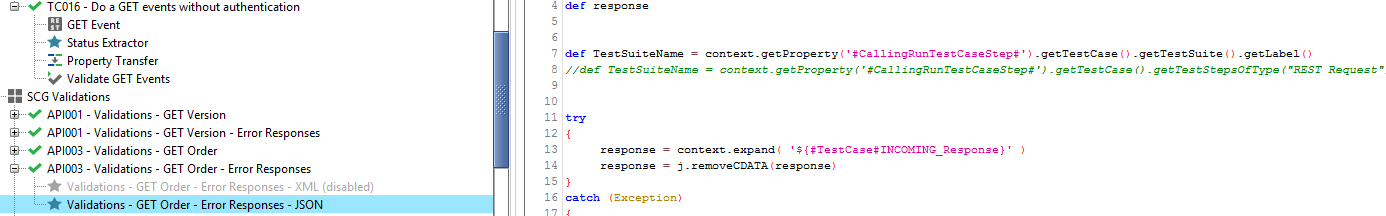
**Referencing back to the Called test case in SOAP UI.**

Below Validate GET Events of TC016 calls API003- Validation- GET Order- Error Responses

Now to reference the Called test case the Context handling can be used like so



This would return the Test Case Name of the calling Test case i.e TC016 – Do a GET events without authentication.



The above concept is called Context Handling.

More info: <https://www.linkedin.com/pulse/soapui-advanced-context-handling-while-invoking-other-nayan-gaur>

**Date Time handling**

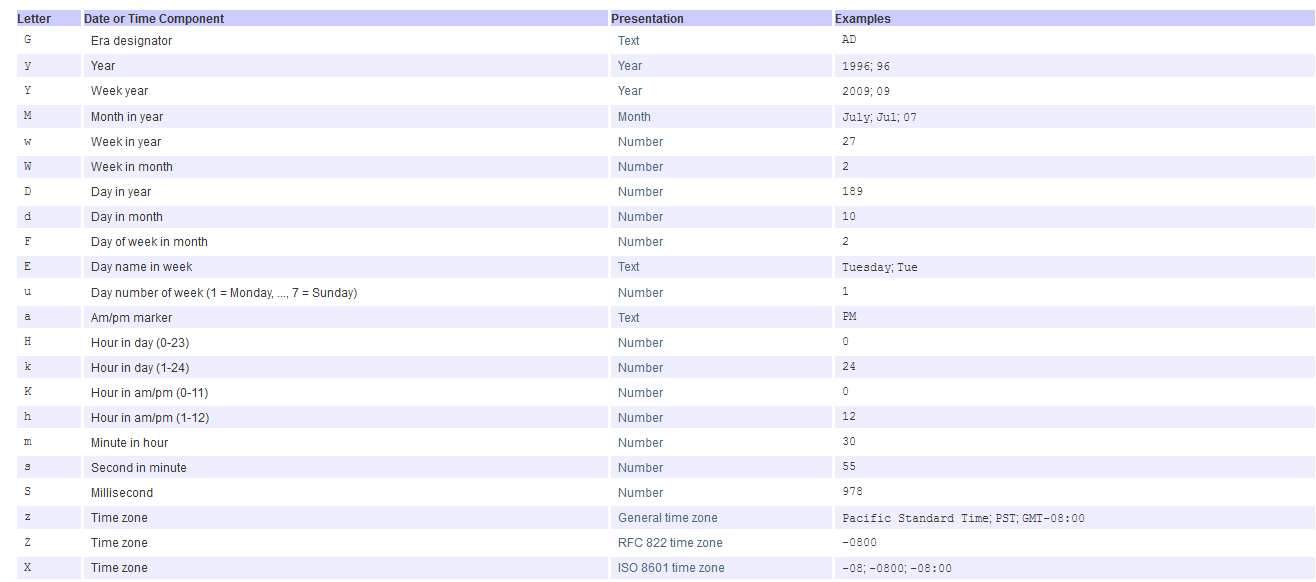
1. To parse DateTime from string into a Date object do the following



Give the format of the Date time as the one appeared in the String.

E.g.: if the String contains “22/07/2017” the code will be as follows

The patter for the format would be as follows



1. To change the format of the date in a **Date object** and to return it as a **Date Object** use :





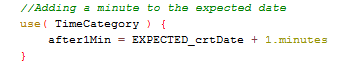
The return type here would be Date Object

1. To change the format of the date in a **Date object** and to return it as a **String Object** use :



The return type would be String Object.

1. Adding time to Date object



Need to  to use the above code.

**Assertion in Groovy**

1. Assertions can be done in Groovy as follows



If errorCount is not 0 then an error message if returned and the execution fails.

A colon ( : ) or a ( , ) both can be used after the condition to write the error message in assert.

**Logging in Groovy**

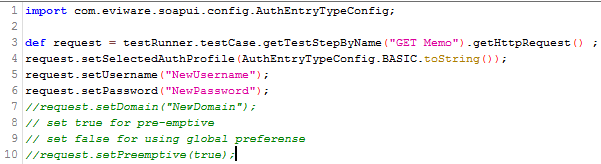
1. Log.info can be used for logging under script log in SOAPUI like so.



This would log it under the Script Log tab in SoapUI.

**Groovy script for setting up Authentication**

This sets up the authentication of the REST request using groovy.



**Fetching RAW request from REST request**



**JSON Library base class with basic methods fo logging errors, comparing strings, comparing list etc.**

