

# Soap UI

# Assertions in SOAPUI

- Contains
- Not Contains
- Xpath Match
- Xpath Query Match
- Invalid HTTP Status codes
- Not SOAP fault
- Schema compliance
- SOAP fault
- SOAP Response
- Valid HTTP Status codes
- Response SLA
- Script Assertion
- **JSON Response for Rest API's**
  - Json Path count
  - JsonPath Existence Match
  - Json path match

# Assertions in SOAPUI(Cont.)

- **Contains** : This assertion checks for the existence of some text in the received response or request message.
- **Not Contains** :The counterpart for the Contains assertion; this one checks that a specified string does not exist in the asserted message.
- **XPath Match** : uses an XPath expression to select content from the target property and compares the result to an expected value. Applicable to any property containing XML.
- **XQuery Match** : uses an XQuery expression to select content from the target property and compares the result to an expected value. Applicable to any property containing XML.
- **Invalid HTTP Status Codes** : checks that the target TestStep received an HTTP result with a status code not in the list of defined codes. Applicable to any TestStep that receives HTTP messages.
- **Valid HTTP Status Codes** : checks that the target TestStep received an HTTP result with a status code in the list of defined codes. Applicable to any TestStep that receives HTTP messages.
- **Not SOAP fault** :checks that the response is not a SOAP Fault.
- **SOAP fault** :checks that the response is a SOAP Fault.
- **Schema compliance** :validates the response message against the definition in the WSDL and contained XML Schema.
- **SOAP Response** : checks that the response is a valid SOAP Response.
- **Response SLA** :This assertion validates that the response time is within the specified value, otherwise the assertion will fail.
- **Script Assertion** : runs a custom script to perform arbitrary validations. Applicable to TestSteps only (i.e. not properties).
- **Json Path count** : counts json paths in JSON response.
- **JsonPath Existence Match** : Checks Json path existence in JSON response
- **Json path match** : Check path in JSON response.

# Assertions in SOAPUI(Cont.)

- WSDL (SOAP)
- <https://footballpool.dataaccess.eu/info.wso?WSDL>
- WADL (REST)
- <https://regres.in/api/users?page=2>
- <http://services.groupkt.com/country/get/iso3code/IND>

# XPath Match Assertion- Examples

//m:AllPlayerNamesResponse/m:AllPlayerNamesResult/m:tPlayerName[1]/m:ild  
1488

//m:AllPlayerNamesResponse/m:AllPlayerNamesResult/m:tPlayerName[1]/m:sName  
Aaron Cresswell

## Count sName tags presence:

count(//m:sName)  
2328

## Tag presence:

//m:AllPlayerNamesResponse/m:AllPlayerNamesResult/m:tPlayerName/m:sName  
true

## Check sName match:

matches(//m:AllPlayerNamesResponse/m:AllPlayerNamesResult/m:tPlayerName[1]/m:sName,"Aaron Cresswell")  
true

## Check Names should have only alphabets:

matches(//m:AllPlayerNamesResponse/m:AllPlayerNamesResult/m:tPlayerName[1]/m:sName,'[A-Za-z]\*')  
true

## Check ID's should have only digits:

matches(//m:AllPlayerNamesResponse/m:AllPlayerNamesResult/m:tPlayerName[1]/m:ild,'\d')  
true

## Check ID's should have only digits range between 2-3:

matches(//m:AllPlayerNamesResponse/m:AllPlayerNamesResult/m:tPlayerName[1]/m:ild,'\d{2,3}')

WSDL :

<https://footballpool.dataaccess.eu/info.wso?WSDL>

More about XPath:

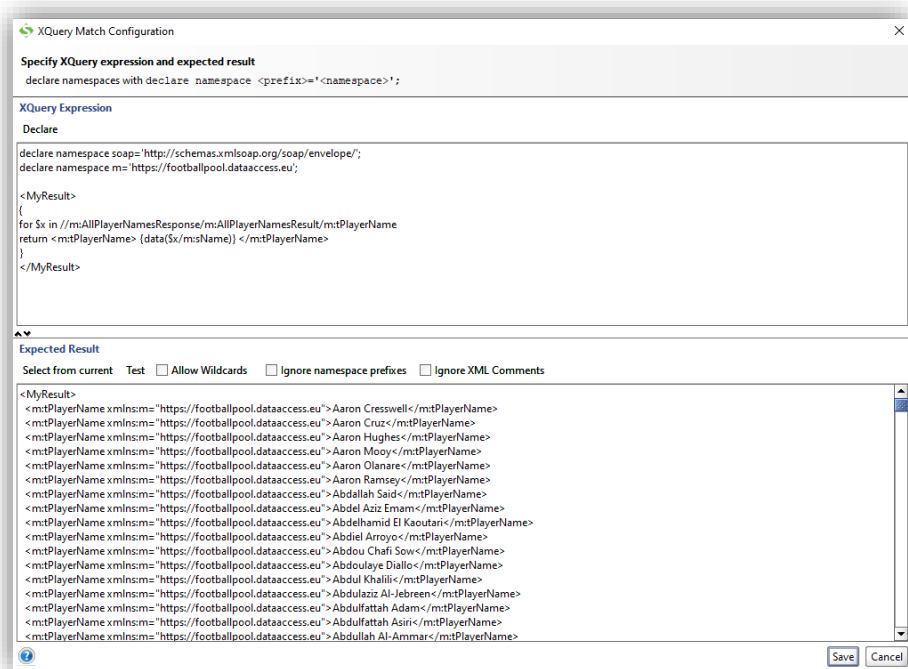
[https://www.w3schools.com/xml/xsl\\_functions.asp](https://www.w3schools.com/xml/xsl_functions.asp)

# XQuery Match

```
<MyResult>
{
for $x in //m:AllPlayerNamesResponse/m:AllPlayerNamesResult/m:tPlayerName
return <m:tPlayerName> {data($x/m:sName)} </m:tPlayerName>
}
</MyResult>
```

WSDL :

<https://footballpool.dataaccess.eu/info.wso?WSDL>



# Script Assertion for XML Response

//check response time

```
assert messageExchange.timeTaken < 4000
```

//check for Endpoint

```
log.info messageExchange.getEndpoint()
```

//check for TimeTaken

```
log.info messageExchange.getTimeTaken()
```

//check for header

```
log.info (messageExchange.responseHeaders["Content-Length"])
```

```
assert messageExchange.responseHeaders["Content-Length"] != null
```

//check for header

```
log.info messageExchange.responseHeaders["Content-Type"]
```

//check for header

```
log.info (messageExchange.responseHeaders["Content-Type"])
```

```
assert messageExchange.responseHeaders["Content-Length"] != null
```

WSDL :

<https://footballpool.dataaccess.eu/info.wso?WSDL>

//check attachments

```
assert messageExchange.responseAttachments.length == 0
```

```
log.info (messageExchange.responseAttachments.length)
```

//validate response nodes

```
def groovyUtils = new com.eviware.soapui.support.GroovyUtils( context )
```

```
def requestHolder = groovyUtils.getXmlHolder( messageExchange.requestContent )
```

```
def responseHolder = groovyUtils.getXmlHolder( messageExchange.responseContent )
```

```
def sName =
```

```
responseHolder.getNodeValue("//m:AllPlayerNamesResponse/m:AllPlayerNamesResult/m  
:tPlayerName[1]/m:sName")
```

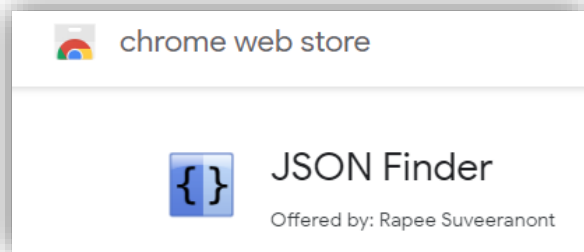
```
assert sName == "Aaron Cresswell"
```

//to get response

```
def resp = messageExchange.responseContentAsXml.toString()
```

# Assertions for Json Response

- JSONPath Finder plugin can be used to get JSON paths for the nodes present in JSON file
- Install JSONPath Finder from chrome web store.



**JSONPath Finder**  
Simple tool that allows you to list all JSONPath expressions for a particular node.

```
{
  "id": 4,
  "first_name": "Eve",
  "last_name": "Holt",
  "avatar": "https://s3.amazonaws.com/ui/faces/twitter/marcoramires/128.jpg"
},
{
  "id": 5,
  "first_name": "Charles",
  "last_name": "Morris",
  "avatar": "https://s3.amazonaws.com/ui/faces/twitter/stephenmoon/128.jpg"
},
{
  "id": 6,
  "first_name": "Tracey",
  "last_name": "Ramos",
  "avatar": "https://s3.amazonaws.com/ui/faces/twitter/blgmancho/128.jpg"
}
}
```

Search Node:

Path Type: ☒ JSONPath ☐ XPath ☐ XMLPath

**JSONPath expressions for : id**

Total JSONPath's found : 3

Path	Result
.data[0].id	4
.data[1].id	5
.data[2].id	6



# Assertions for Json Response (Cont.)

WADL (REST)

<https://regres.in/api/users?page=2>

JSONPath Count Match Configuration

Specify JSONPath expression and expected result

JSONPath Expression

.data[0].first\_name

JsonPath Count

Expected Result

Select from current Test

1

JSONPath Match Configuration

Specify JSONPath expression and expected result

JSONPath Expression

.data[0].first\_name

JsonPath Match

Expected Result

Select from current Test

Eve

# Assertions for Json Response (Cont.)

WADL (REST)

<https://reqres.in/api/users?page=2>

JSONPath Existence Match Configuration

Specify JSONPath expression and expected result

JSONPath Expression

.data[0].first\_name

JsonPath Existence Match

Expected Result

Select from current Test

true

JSONPath RegEx Match Configuration

Specify JSONPath expression and expected result

JSONPath Expression

.data[1].first\_name

JsonPath RegEx Match

Regular Expression

.\*Char.\*

Expected Result

Select from current Test

true

# Script Assertion for JSON Response

```
//get json response
```

```
import groovy.json.JsonSlurper
```

```
def responseMessage = messageExchange.response.responseContent
```

```
def json = new JsonSlurper().parseText(responseMessage)
```

```
//assert node values
```

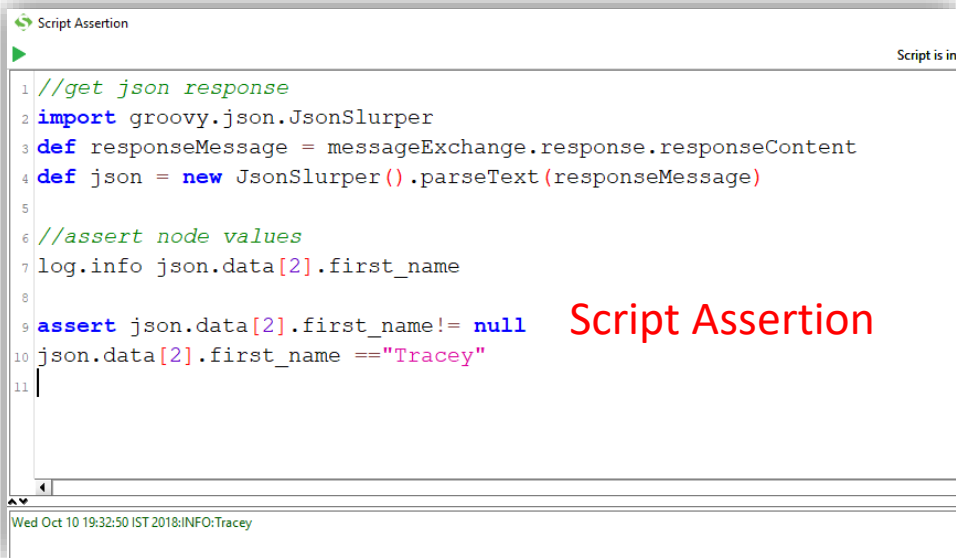
```
log.info json.data[2].first_name
```

```
assert json.data[2].first_name!= null
```

```
json.data[2].first_name == "Tracey"
```

WADL (REST)

<https://reqres.in/api/users?page=2>



```
Script Assertion
Script is in

1 //get json response
2 import groovy.json.JsonSlurper
3 def responseMessage = messageExchange.response.responseContent
4 def json = new JsonSlurper().parseText(responseMessage)
5
6 //assert node values
7 log.info json.data[2].first_name
8
9 assert json.data[2].first_name!= null
10 json.data[2].first_name == "Tracey"
11

Wed Oct 10 19:32:50 IST 2018:INFO:Tracey
```

Script Assertion