

SoapUI

Properties in SOAPUI

1. What is property in soapui
 2. Why do we use it
 3. How to create properties at different levels
 4. How to refer properties
- Properties can be used as variables to store values that can be referred in testing
 - Properties can be accessed at following levels:

Project - \${#Project#PropertyName}
TestSuite - \${#TestSuite#PropertyName}
TestCase - \${#TestCase#PropertyName}
TestStep - \${TestStepName#PropertyName}
System - \${#System#PropertyName}
Env - \${#Env#PropertyName}
Global - \${#Global#PropertyName}

Properties in SOAPUI (Cont.)

- **WSDL** : <https://footballpool.dataaccess.eu/info.wso?WSDL>
- **Test Case**: AllPlayersWithRole



The screenshot displays the SOAPUI interface for a test case named "AllPlayersWithRole". The URL bar shows "https://footballpool.dataaccess.eu/info.wso". The left pane shows the XML request, and the right pane shows the XML response.

Request XML:

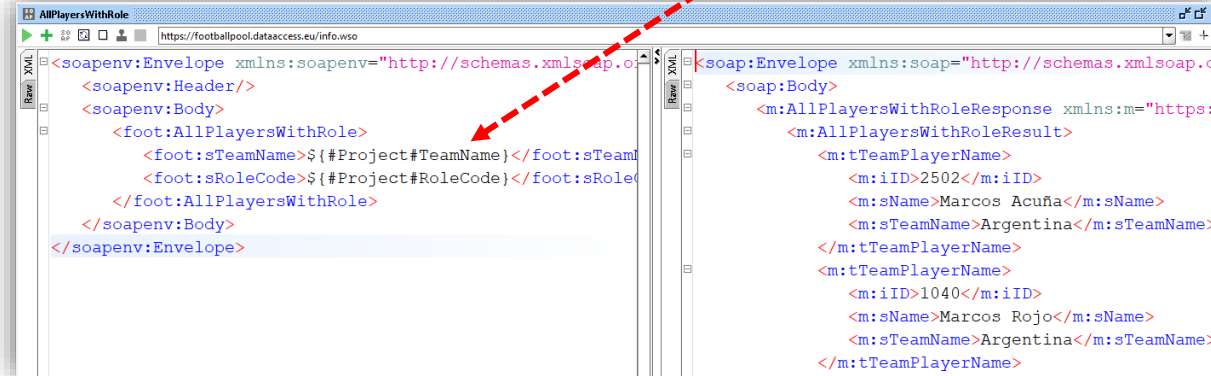
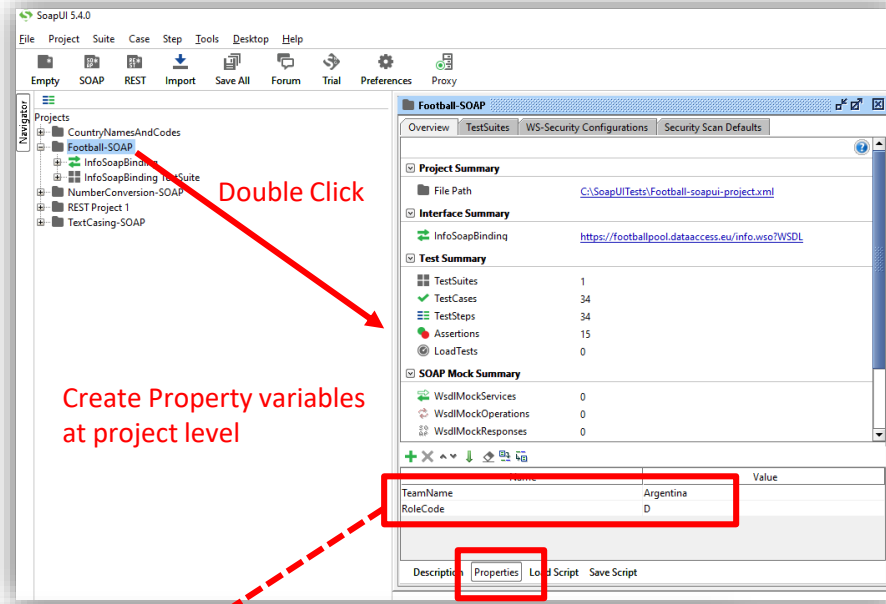
```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Header/>
  <soapenv:Body>
    <foot:AllPlayersWithRole>
      <foot:sTeamName>Argentina</foot:sTeamName>
      <foot:sRoleCode>D</foot:sRoleCode>
    </foot:AllPlayersWithRole>
  </soapenv:Body>
</soapenv:Envelope>
```

The request XML is highlighted in blue, and the `<foot:sTeamName>Argentina</foot:sTeamName>` and `<foot:sRoleCode>D</foot:sRoleCode>` elements are enclosed in a red box.

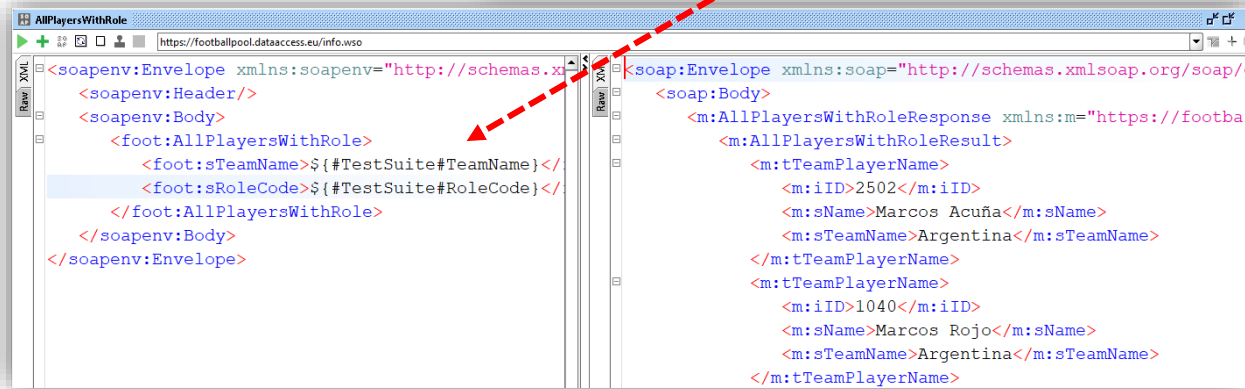
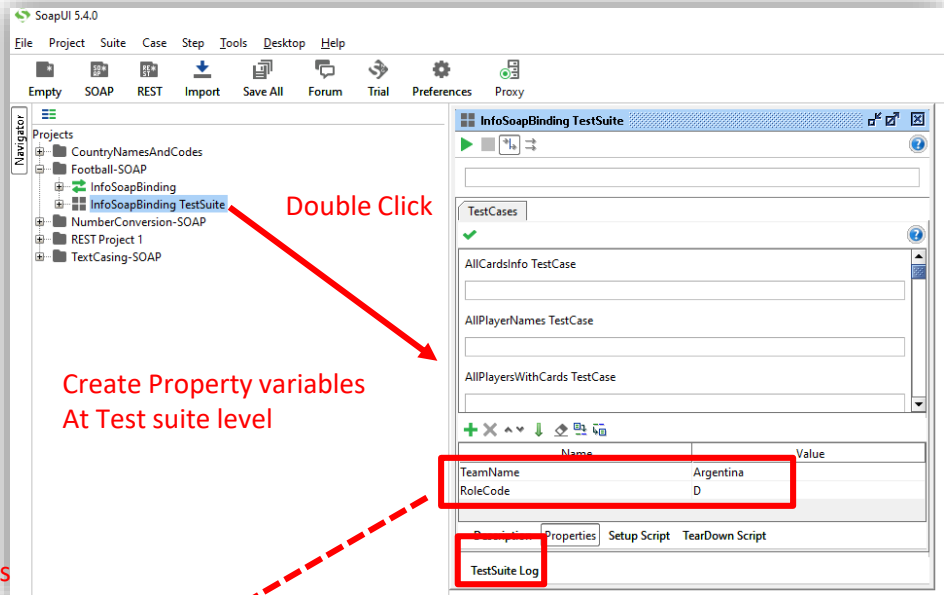
Response XML:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <m:AllPlayersWithRoleResponse xmlns:m="https://footballpool.dataaccess.eu/info.wso">
      <m:AllPlayersWithRoleResult>
        <m:tTeamPlayerName>
          <m:iID>2502</m:iID>
          <m:sName>Marcos Acuña</m:sName>
          <m:sTeamName>Argentina</m:sTeamName>
        </m:tTeamPlayerName>
        <m:tTeamPlayerName>
          <m:iID>1040</m:iID>
          <m:sName>Marcos Rojo</m:sName>
          <m:sTeamName>Argentina</m:sTeamName>
        </m:tTeamPlayerName>
      </m:AllPlayersWithRoleResult>
    </m:AllPlayersWithRoleResponse>
  </soap:Body>
</soap:Envelope>
```

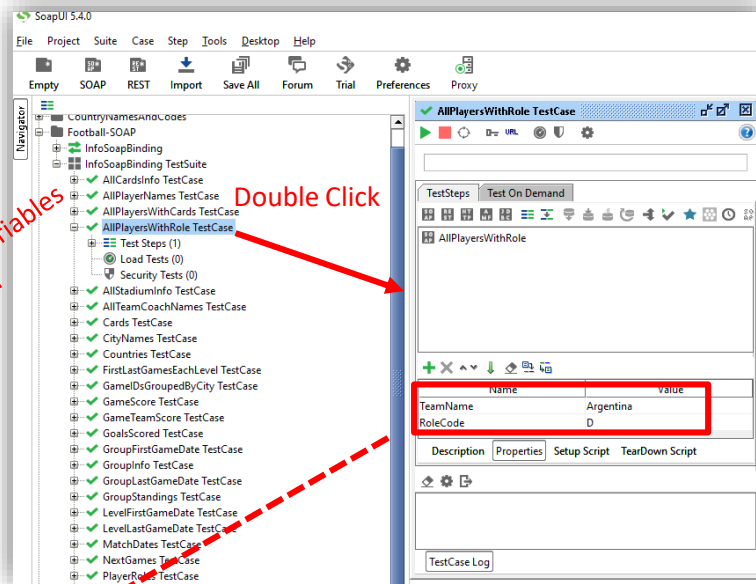
Project Level



Test Suite Level

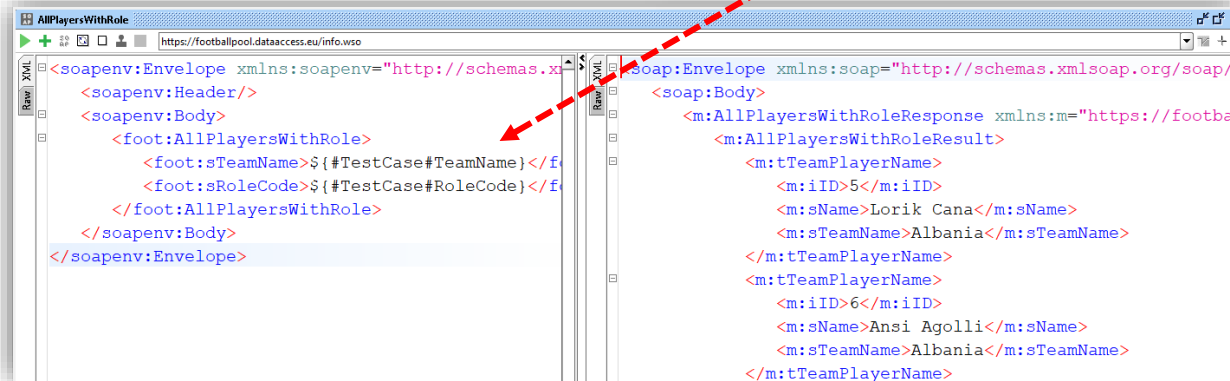


Test Case Level

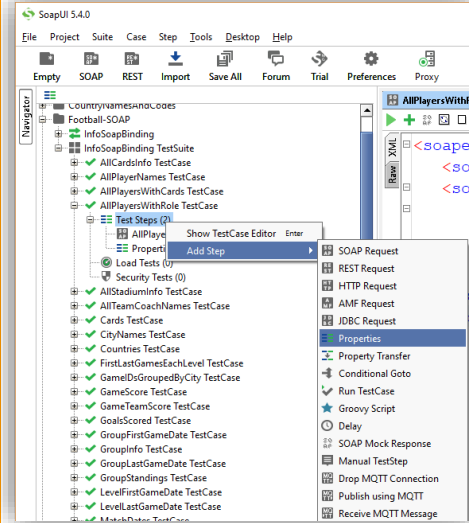


Create Property variables
at Test case level

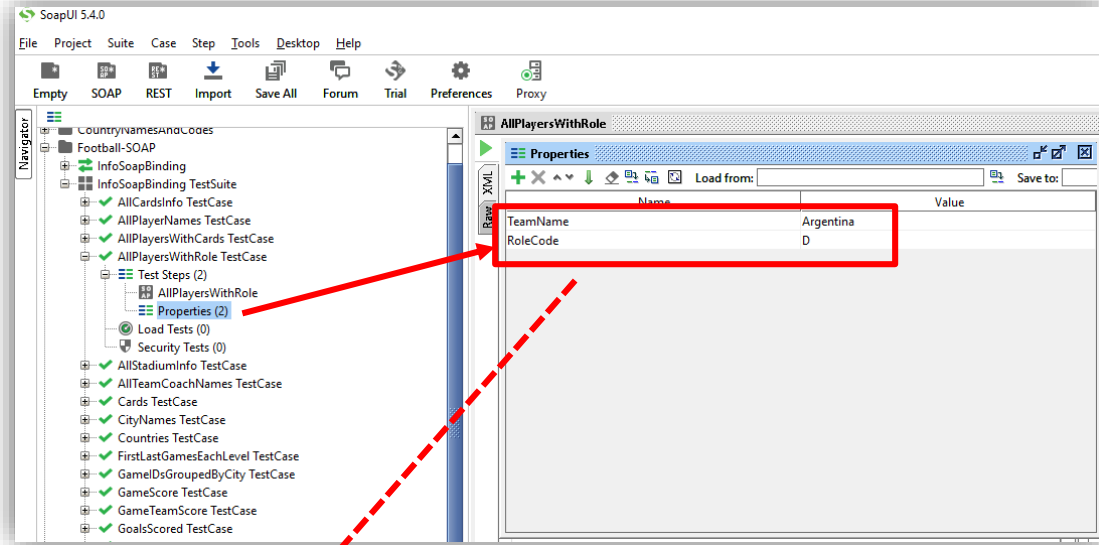
Usage of Property variables



Test Step Level



Add step → Properties

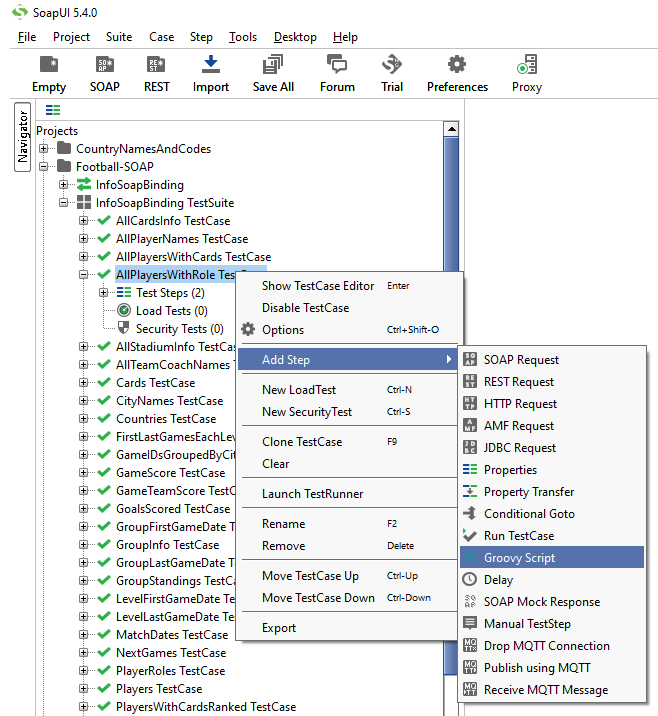


Usage of Property variables



Access property values using groovy script

- `def message1=context.expand('${#Project#msg}')`
- `log.info message1`
- `def message2=context.expand('${#TestSuite#msg}')`
- `log.info message2`
- `def message3=context.expand('${#TestCase#msg}')`
- `log.info message3`
- `def message4=context.expand('${Properties#msg}')`
- `log.info message4`



Access property values using groovy script(Cont.)

SoapUI 5.4.0

File Project Suite Case Step Tools Desktop Help

Empty SOAP REST Import Save All Forum Trial Preferences Proxy

Navigator

Projects

- CountryNamesAndCodes
- Football-SOAP
 - InfoSoapBinding
 - InfoSoapBinding TestSuite
 - AllCardsInfo TestCase
 - AllPlayerNames TestCase
 - AllPlayersWithCards TestCase
 - AllPlayersWithRole TestCase
 - Test Steps (3)
 - Properties (3)
 - Groovy Script
 - AllPlayersWithRole
 - Load Tests (0)
 - Security Tests (0)
 - AllStadiumInfo TestCase
 - AllTeamCoachNames TestCase
 - Cards TestCase
 - CityNames TestCase
 - Countries TestCase
 - FirstLastGamesEachLevel TestCase
 - GameIdsGroupedByCity TestCase
 - GameScore TestCase
 - GameTeamScore TestCase
 - GoalsScored TestCase
 - GroupFirstGameDate TestCase
 - GroupInfo TestCase
 - GroupLastGameDate TestCase
 - GroupStandings TestCase
 - LevelFirstGameDate TestCase
 - LevelLastGameDate TestCase
 - MatchDates TestCase
 - NextGames TestCase

Groovy Script

Script is invoked with log, context and testRunner variables

```
1 def message1=context.expand('${#Project#msg}')
2 log.info message1
3
4 def message2=context.expand('${#TestSuite#msg}')
5 log.info message2
6
7 def message3=context.expand('${#TestCase#msg}')
8 log.info message3
9
10 def message4=context.expand('${#Properties#msg}')
11 log.info message4
12
13
```

Thu Oct 11 14:25:12 IST 2018:INFO: am Project level property
Thu Oct 11 14:25:12 IST 2018:INFO: am Test suite level property
Thu Oct 11 14:25:12 IST 2018:INFO: am Test case level property
Thu Oct 11 14:25:12 IST 2018:INFO: am Test step level property

Log Output (4)

11:18

Property Transfer | Values Transfer

The screenshot displays two XML responses in SoapUI:

StadiumNames Response:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Header/>
  <soapenv:Body>
    <foot:StadiumNames/>
  </soapenv:Body>
</soapenv:Envelope>
```

StadiumInfo Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Header/>
  <soapenv:Body>
    <foot:StadiumInfo>
      <foot:sStadiumName>Cosmos Arena</foot:sStadiumName>
    </foot:StadiumInfo>
  </soapenv:Body>
</soapenv:Envelope>
```

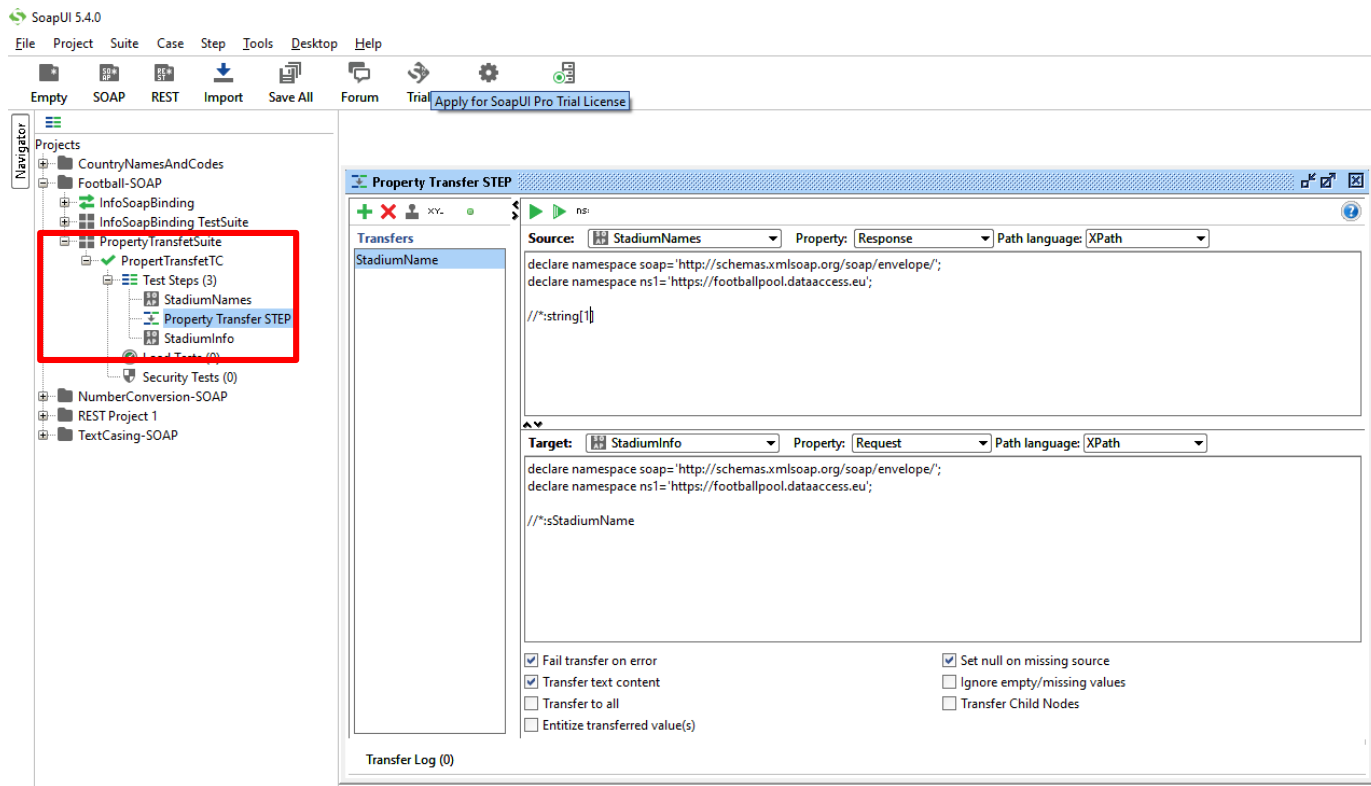
StadiumInfo Response:

```
<soapenv:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <m:StadiumInfoResponse xmlns:m="https://footballpool.dataaccess.eu/info">
      <m:StadiumInfoResult>
        <m:sName>Cosmos Arena</m:sName>
        <m:iSeatsCapacity>44918</m:iSeatsCapacity>
        <m:sCityName>Samara</m:sCityName>
        <m:sWikipediaURL>https://en.wikipedia.org/wiki/Cosmos_Arena</m:sWikipediaURL>
        <m:sGoogleMapsURL>https://www.google.com/maps/place/Cosmos+Arena</m:sGoogleMapsURL>
      </m:StadiumInfoResult>
    </m:StadiumInfoResponse>
  </soap:Body>
</soap:Envelope>
```

Steps in One test case:

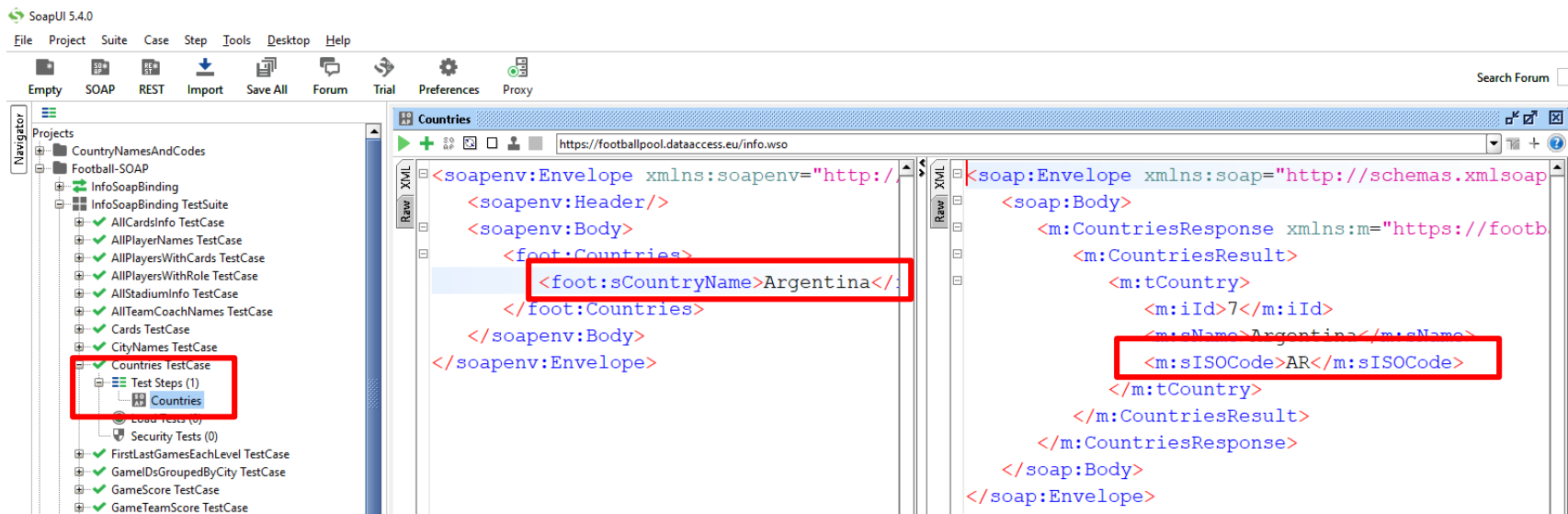
- StadiumNames
- StadiumInfo

Property Transfer Step




Parameterization/Data Driven Testing

- **Test Case:** Countries Test Case



Parameterization/DDT –Data Source as Excel

DataSource-XLS

 **Properties**

CountryName

[Learn about data sources](#)

Configuration

DataSource: Excel

File: C:/SoapUITests/Countries.xlsx [Browse...](#)

Worksheet: Sheet1

Start at Cell: A2

Ignore Empty: ☒ Select to skip empty rows

Evaluate formulas: ☐ Select if the file contains formulas that should be evaluated

CountryName
Argentina
Australia
Belgium
Brazil
Colombia
Croatia

Parameterization/DDT –Data Source as Text File

DataSource-TextFile

Properties

CountryName

Configuration

DataSource: File

File: C:/SoapUITests/Countries.txt [Browse...](#)

Separator: ,

Charset: Cp1252

Trim: ☒ Trim values

Quoted Values: ☐ Values are in quotes

[Learn about data sources](#)

CountryName
Argentina
Australia
Belgium
Brazil
Colombia
Croatia

Parameterization/DDT –Data Generator

DataSource-DataGeneration

+ - X

↓

XY

▲ ▼

Properties

CountryName

▶ ⚙

DataSource: Data Generator ▼

Configuration

Number of rows: 10

Property type: Country ▼

Duplicate each value: ☐ 1 time(s)

Sample:

Barbados
Cayman Islands
Palestinian Territory, Occupied

Set Default Value

[Learn about data sources](#)

CountryName
Djibouti
Taiwan, Province of China
Bulgaria
Serbia and Montenegro
Zimbabwe
Christmas Island
Japan
Mayotte
France
Holy See (Vatican City State)

www.pavanonlinetrainings.com