//Validation Part

//get the response from 'get'

def getRes=testRunner.testCase.testSuite.testCases["getEmployee"].testSteps["get"].getPropertyValue("Response")

//for both set property or get property to get or set xml, we have to rely on XmlHolder

def getEmployeeRes=new XmlHolder(getRes)

//now get the node value

//\*\*\*\*\*\*the response\*\*\*\*\*\*\*\*

//<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">

// <soapenv:Body>

// <ns:getEmployeeDetailsResponse xmlns:ns="http://sample.com/reservation/guest/types">

// <ns:return xsi:type="ns:Employee" xmlns:ax21="http://exception.reservation.sample.com/xsd" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

// <ns:age>30</ns:age>

// <ns:department>cse</ns:department>

// <ns:id>90</ns:id>

// <ns:name>Sun</ns:name>

// </ns:return>

// </ns:getEmployeeDetailsResponse>

// </soapenv:Body>

//</soapenv:Envelope>

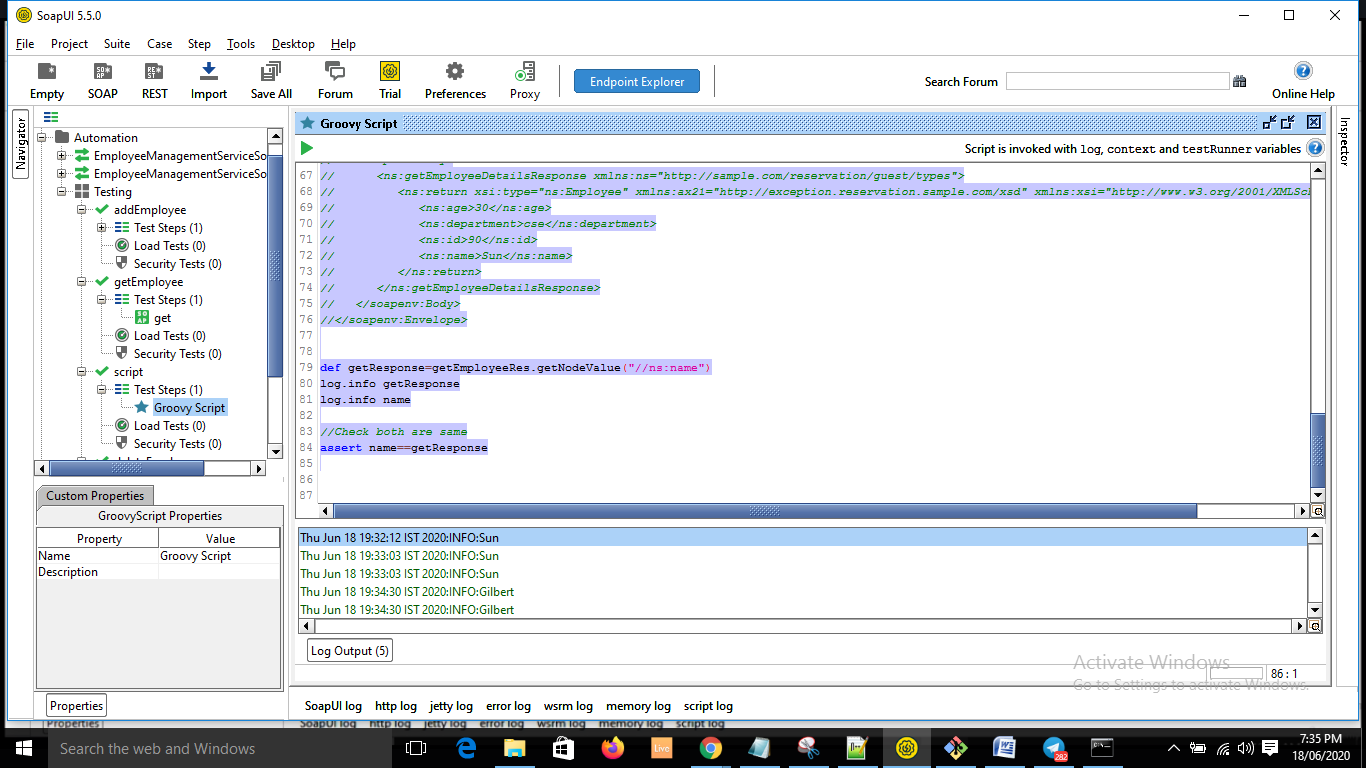
def getResponse=getEmployeeRes.getNodeValue("//ns:name")

log.info getResponse

log.info name

//Check both are same

assert name==getResponse



Assignment:

Put the added employee iin deleteEmployee and delete it.

Full code as of now:

import com.eviware.soapui.support.XmlHolder

import com.eviware.soapui.impl.wsdl.testcase.WsdlTestRunContext

//Hit AddEmployee Request with properties from Test Case level.

//Pull the response stored in test script:

def addReq=testRunner.testCase.testSuite.testCases["addEmployee"].testSteps["add"].getPropertyValue("Request")

def xmlAdd=new XmlHolder(addReq)

def name=testRunner.testCase.testSuite.testCases["addEmployee"].getPropertyValue("name")

def id=testRunner.testCase.testSuite.testCases["addEmployee"].getPropertyValue("id")

def age=testRunner.testCase.testSuite.testCases["addEmployee"].getPropertyValue("age")

xmlAdd.setNodeValue("//typ:addEmployee/typ:name", name)

xmlAdd.setNodeValue("//typ:addEmployee/typ:id", id)

xmlAdd.setNodeValue("//typ:addEmployee/typ:Department", "cse")

xmlAdd.setNodeValue("//typ:addEmployee/typ:age", age)

def newAddXml= xmlAdd.getXml()

//log.info newAddXml

testRunner.testCase.testSuite.testCases["addEmployee"].testSteps["add"].setPropertyValue("Request",newAddXml)

//call the service

//context of addEmployee test case should be passed as Add step which we are trying to run belongs to that test case.

def addTestStep=testRunner.testCase.testSuite.testCases["addEmployee"].testSteps["add"]

def contextAddEmployee=new WsdlTestRunContext(addTestStep)

addTestStep.run(testRunner,contextAddEmployee)

//Hit getEmployee and generate response, validate the response if it have the name which added in our earlier addEmployee.

//1. go to the getEmployee and get the property value for Request.

def getReq=testRunner.testCase.testSuite.testCases["getEmployee"].testSteps["get"].getPropertyValue("Request")

//2. load the request xml in XmlHolder.

def getEmpxml=new XmlHolder(getReq)

//3. put the value in correct xml node.

getEmpxml.setNodeValue("//typ:getEmployeeDetails/typ:employeeName",name)

//4. save the xml now

def newGetEmpXml=getEmpxml.getXml()

//5. place it back in the set property.

testRunner.testCase.testSuite.testCases["getEmployee"].testSteps["get"].setPropertyValue("Request",newGetEmpXml)

//6. now reach get service test step control

def getTestStep=testRunner.testCase.testSuite.testCases["getEmployee"].testSteps["get"]

//we have to create a context specific to the test case. For that pass a test step beloning to the test case to WsdlTestRunContext()

def contextGetEmployee= new WsdlTestRunContext(getTestStep)

getTestStep.run(testRunner,contextGetEmployee)

//Validation Part

//get the response from 'get'

def getRes=testRunner.testCase.testSuite.testCases["getEmployee"].testSteps["get"].getPropertyValue("Response")

//for both set property or get property to get or set xml, we have to rely on XmlHolder

def getEmployeeRes=new XmlHolder(getRes)

//now get the node value

def getResponse=getEmployeeRes.getNodeValue("//ns:name")

log.info getResponse

log.info name

//Check both are same

assert name==getResponse

* Code from the trainer

import com.eviware.soapui.support.XmlHolder  
import com.eviware.soapui.impl.wsdl.testcase.WsdlTestRunContext

log.info "Hello this is my first program"

//Context variable is used to access and modify the properties of your testcase  
//rahulonlinetutor@gmail.com  
//scope of context lies in Testcase only!

log.info context.expand('${#TestCase#name}')

log.info context.expand('${#TestCase#age}')  
// To access/modify any property you have to come through its parent only  
//Project->Test Suite -> Test Cases\_> TestStep  
log.info testRunner.testCase.testSuite.testCases["GetEmployee"].getPropertyValue("id")  
 testRunner.testCase.testSuite.testCases["GetEmployee"].setPropertyValue("id","123")  
log.info testRunner.testCase.testSuite.testCases["AddEmployee"].getPropertyValue("age")

log.info testRunner.testCase.getPropertyValue("dept")

log.info testRunner.testCase.testSuite.testCases["GetEmployee"].testSteps["get"].getPropertyValue("Request")

def project=testRunner.testCase.testSuite.project  
project.getPropertyValue("gender")

//Hit Add employee Request with properties from Testcase level.

def addReq=testRunner.testCase.testSuite.testCases["AddEmployee"].testSteps["add"].getPropertyValue("Request")  
def name=testRunner.testCase.testSuite.testCases["AddEmployee"].getPropertyValue("name")  
def id=testRunner.testCase.testSuite.testCases["AddEmployee"].getPropertyValue("id")  
def age=testRunner.testCase.testSuite.testCases["AddEmployee"].getPropertyValue("age")  
 def xmlAdd= new XmlHolder(addReq) //

 xmlAdd.setNodeValue("//typ:addEmployee/typ:name",name)  
   
 xmlAdd.setNodeValue("//typ:addEmployee/typ:id",id)  
  xmlAdd.setNodeValue("//typ:addEmployee/typ:Department","cse")  
   xmlAdd.setNodeValue("//typ:addEmployee/typ:age",age)  
  def newAddXml= xmlAdd.getXml();

  testRunner.testCase.testSuite.testCases["AddEmployee"].testSteps["add"].setPropertyValue("Request",newAddXml)  
  //log.info newAddXml  
   
 //call the service-  
 //context of AddEmployee testcase should be passed as Add step which we are trying to run belongs to that Testcase

def addTestStep=testRunner.testCase.testSuite.testCases["AddEmployee"].testSteps["add"]  
def contextAddEmployee= new WsdlTestRunContext(addTestStep);  
   
 addTestStep.run(testRunner,contextAddEmployee)

 //Hit GetEmployee and generate response..Validate the response if it have the name which added in our earlier Step

 def getReq=testRunner.testCase.testSuite.testCases["GetEmployee"].testSteps["get"].getPropertyValue("Request")  
def getEmpxml= new XmlHolder(getReq)  
 getEmpxml.setNodeValue("//typ:getEmployeeDetails/typ:employeeName",name)  
 def newgetEmpxml=getEmpxml.getXml()  
testRunner.testCase.testSuite.testCases["GetEmployee"].testSteps["get"].setPropertyValue("Request",newgetEmpxml)

def getTestStep= testRunner.testCase.testSuite.testCases["GetEmployee"].testSteps["get"];

def contextGetEmployee= new WsdlTestRunContext(getTestStep);  
getTestStep.run(testRunner,contextGetEmployee)

//Validation -  
 def getRes=testRunner.testCase.testSuite.testCases["GetEmployee"].testSteps["get"].getPropertyValue("Response")  
   
def getEmpRes= new XmlHolder(getRes)  
def Getresponse= getEmpRes.getNodeValue("//ns:name")  
log.info Getresponse

log.info name

assert Getresponse==name

//Delete -