**Webservice API testing using SOAP UI and automation using Groovy script**

API URL: <https://mindtree-emp.azurewebsites.net/swagger-ui.html#/employee-controller>

1. Go through the above URL to get details about the APIs.
2. Create a rest project using the above rest service API.

Graphical user interface, text, application, email

Description automatically generated

1. Add test suite, test case and test step for testing webservice API for GET, POST, PUT & DELETE requests

Graphical user interface, text

Description automatically generated

Graphical user interface, text

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application

Description automatically generated

1. Design a groovy script to read the data for competency, id, name and yearOfJoining from excel file and perform the below operation
   1. Use the data we read in step-3 to process the POST request.
   2. Use the GET request (https://mindtree-emp.azurewebsites.net/api/v1/employees/<empID>) within the groovy script to verify the employee data that we added in step – 3a.
   3. Use PUT request to update the data for particular employee and verify with the GET request.
   4. Use DELETE request to delete an employee by reading the empID from the excel file and verify with GET request to confirm if the employee is deleted.
   5. Make use of assertions to verify the response.

Graphical user interface, text, email

Description automatically generated

Text, email

Description automatically generated

Graphical user interface, text, email

Description automatically generated

Text, email

Description automatically generated

Graphical user interface, text, email

Description automatically generated

Graphical user interface, text, email

Description automatically generated

Graphical user interface, text, email

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

Groovy Script:

import jxl.\*

import jxl.write.\*

import groovy.json.JsonSlurper

//Excel Data For POST

Workbook work = Workbook.getWorkbook(new File("C://Users//M1080645//Desktop//EmployeeData.xls"));

Sheet data = work.getSheet(0)

String compitency = data.getCell(0,1).getContents()

String id = data.getCell(1,1).getContents()

String name = data.getCell(2,1).getContents()

String yearOfJoining = data.getCell(3,1).getContents()

//Post

testRunner.testCase.testSuite.setPropertyValue("compitency",compitency)

testRunner.testCase.testSuite.setPropertyValue("id",id)

testRunner.testCase.testSuite.setPropertyValue("name",name)

testRunner.testCase.testSuite.setPropertyValue("yearOfJoining",yearOfJoining)

def tCase = testRunner.testCase.testSuite.testCases["TestCase 1"]

def getIdTestStep = tCase.testSteps["POST Request"]

log.info"Running POST Request"

def restRequest=getIdTestStep.run(testRunner,context)

def responseJson =getIdTestStep.testRequest.response.responseContent

JsonSlurper slurper = new JsonSlurper()

def parsedJson = slurper.parseText(responseJson)

log.info(responseJson)

if(parsedJson.httpStatus.equals("CREATED")){

log.info"POST Request has run successfully"

assert true

}else{

log.info"POST was unsuccessful"

assert false

}

//Get

testRunner.testCase.testSuite.setPropertyValue("id",id)

def getIdTestStep1 = tCase.testSteps["GET Request"]

log.info"Running GET Request"

def restRequest1=getIdTestStep1.run(testRunner,context)

def responseJson1 =getIdTestStep1.testRequest.response.responseContent

def parsedJson1 = slurper.parseText(responseJson1)

log.info(responseJson1)

if(parsedJson1.httpStatus.equals("ACCEPTED")){

log.info"GET Request has run successfully"

assert true

}

log.info(compitency)

if(parsedJson1.body.name.equals(name) && parsedJson1.body.compitency.equals(compitency) && parsedJson1.body.yearOfJoining.equals(yearOfJoining.toInteger())){

log.info"New Employee was added"

assert true

}else{

log.info"Employee was not added"

assert false

}

//Excel Data For PUT

compitancy = data.getCell(0,2).getContents()

id = data.getCell(1,2).getContents()

name = data.getCell(2,2).getContents()

yearOfJoining = data.getCell(3,2).getContents()

//Put

testRunner.testCase.testSuite.setPropertyValue("compitency",compitency)

testRunner.testCase.testSuite.setPropertyValue("id",id)

testRunner.testCase.testSuite.setPropertyValue("name",name)

testRunner.testCase.testSuite.setPropertyValue("yearOfJoining",yearOfJoining)

log.info"Running PUT Request"

def getIdTestStep2 = tCase.testSteps["PUT Request"]

def restRequest2=getIdTestStep2.run(testRunner,context)

def responseJson2 =getIdTestStep2.testRequest.response.responseContent

def parsedJson2 = slurper.parseText(responseJson2)

log.info(responseJson2)

if(parsedJson2.httpStatus.equals("UPDATED")){

log.info"PUT Request has run successfully"

assert true

}else{

log.info"PUT was unsuccessful"

assert false

}

//Get

testRunner.testCase.testSuite.setPropertyValue("id",id)

def getIdTestStep3 = tCase.testSteps["GET Request"]

def restRequest3=getIdTestStep3.run(testRunner,context)

log.info"Running GET Request"

def responseJson3 =getIdTestStep3.testRequest.response.responseContent

def parsedJson3 = slurper.parseText(responseJson3)

log.info(responseJson3)

if(parsedJson3.httpStatus.equals("ACCEPTED")){

log.info"GET Request has run successfully"

assert true

}

if(parsedJson3.body.name.equals(name) && parsedJson3.body.compitency.equals(compitency) && parsedJson3.body.yearOfJoining.equals(yearOfJoining.toInteger())){

log.info"Employee was Updated"

assert true

}else{

log.info"Employee was not Updated"

assert false

}

//Delete

testRunner.testCase.testSuite.setPropertyValue("id",id)

def getIdTestStep4 = tCase.testSteps["DELETE Request"]

log.info"Running DELETE Request"

def restRequest4=getIdTestStep4.run(testRunner,context)

def responseJson4 =getIdTestStep4.testRequest.response.responseContent

def parsedJson4 = slurper.parseText(responseJson4)

log.info(responseJson4)

if(parsedJson4.httpStatus.equals("DELETED")){

log.info"DELETE Request has run successfully"

assert true

}else{

log.info"DELETE was unsuccessful"

assert false

}

//Get

testRunner.testCase.testSuite.setPropertyValue("id",id)

def getIdTestStep5 = tCase.testSteps["GET Request"]

log.info"Running GET Request"

def restRequest5=getIdTestStep5.run(testRunner,context)

def responseJson5 =getIdTestStep5.testRequest.response.responseContent

def parsedJson5 = slurper.parseText(responseJson5)

log.info(responseJson5)

if(parsedJson5.status.equals(500)){

log.info"GET Request has run successfully"

log.info"Employee was deleted"

assert true

}else{

log.info"Employee was not deleted"

assert false

}

work.close()

final output:

testcase level:

Graphical user interface, text, application, email

Description automatically generated