3.1 GET, POST, XML, and JSON



This section will guide you to know:

* About the problem statement on GET, POST, XML, and JSON
* Solution for the problem statement

**Development Environment:**

* Java 1.8
* Eclipse
* TestNG
* Maven

This guide has three subsections, namely:

3.1.1 Problem statement for GET, POST, XML, and JSON

3.1.2 Solution for the problem statement

3.1.3 Pushing the code to GitHub repositories

**Step 3.1.1:** Problem statement for GET, POST, XML, and JSON

* **Objective:** As a part of developing a functionality, get the list of employees in a particular organization based on the number of parameters passed.
* **Following requirements should be met for problem statement:**
* Create a Maven project.
* Create a JSON package inside the Maven project.
* Create a class inside the package.
* Create a method for GET, POST, and XML to validate the response code from API for using REST Assured.
* Create a JSON Object.
* Create an HTTP request.

**Step 3.1.2:** Solution for the problem statement

Write the program for GET request, POST request, and XML payload using REST API in REST Assured and click on Save.

package Response;

**import** **static** io.restassured.RestAssured.given;

**import** org.json.simple.JSONObject;

**import** org.testng.annotations.Test;

**import** io.restassured.RestAssured;

**import** io.restassured.http.ContentType;

**import** io.restassured.http.Method; **import** io.restassured.response.Response;

**import** io.restassured.specification.RequestSpecification;

**import** junit.framework.Assert;

public class PostResponse {

@Test

**void** RegistrationSuccessful()

{

//specify base URI

RestAssured.baseURI="http://192.168.1.207:8080/api/employee/";

//Request object

RequestSpecification httpRequest=RestAssured.given();

//contains the information in the json format

//Request payload sending along with post request

JSONObject requestParams=new JSONObject();

requestParams.put("empName","Lavanya");

requestParams.put("empAddress","Bomanahalli");

requestParams.put("mobileNumber","9119321102");

requestParams.put("department","testing");

requestParams.put("teamLead","Aruna");

requestParams.put("salary","10000");

requestParams.put("joiningDate","14-05-19");

httpRequest.header("Content-Type", "application/json");

httpRequest.body(requestParams.toJSONString());//attach above data to the request

//Response object

Response response=httpRequest.request(Method.POST,"/add");

//print response in console window

**String** responseBody=response.getBody().asString();

System.out.println("Response Body is:" +responseBody);

//status code validation

**int** statusCode=response.getStatusCode();

System.out.println("Status code is:" +statusCode);

Assert.assertEquals(200,statusCode);

}

@Test

**void** getempDetails()

{

//specify baseUrI

RestAssured.baseURI="http://192.168.1.207:8080/api/employee/search";

//Request object

RequestSpecification httpRequest=RestAssured.given();

//Response object

Response response=httpRequest.request(Method.GET,"/8970922880");

//print response in console window

**String** responseBody=response.getBody().asString();

System.out.println("Response Body is:" +responseBody);

//status code validation

**int** statusCode=response.getStatusCode();

System.out.println("status code is :"+statusCode);

Assert.assertEquals(200,statusCode);

}

@Test

public **void** post\_xml\_test() {

Response response = given().contentType(ContentType.XML).accept(ContentType.XML).body("<Employee>" +

"<empName>Lavanya Gowda</empName>" + "<empAddress>abc</empAddress>"

+ "<mobileNumber>1592210060</mobileNumber>" + "<department>abc</department>" + "<project>abc</project>"

+ "<teamLead>abc</teamLead>" + "<salary>10000</salary>" + "<joiningDate>11-10-19</joiningDate>"

+ "</Employee>").when().post("http://192.168.1.207:8080/api/employee/add/xml");

System.out.println("POST Response\n" + response.asString());

// tests

**int** statusCode = response.getStatusCode();

System.out.println("Status code is:" + statusCode);

Assert.assertEquals(200, statusCode);

}

}

* Click on run and check the output in TestNG.

[RemoteTestNG] detected TestNG version 6.14.3

Response Body is:88

Status code is:200

Response Body is:{"empId":5,"empName":"Lavanya Gowda","empAddress":"bomanahalii","mobileNumber":8970922880,"department":"testing","project":"Cervical cancer application","teamLead":"Aruna","salary":10000.0,"joiningDate":"14-05-19"}

status code is :200

POST Response

89

Status code is:200

PASSED: RegistrationSuccessful

PASSED: getempDetails

PASSED: post\_xml\_test

===============================================

Default test

Tests run: 3, Failures: 0, Skips: 0

===============================================

===============================================

Default suite

Total tests run: 3, Failures: 0, Skips: 0

===============================================

**Step 3.1.3:** Pushing the code to GitHub repositories

Open your command prompt and navigate to the folder where you have created your files.

cd <folder path>

Initialize your repository using the following command:

git init

Add all the files to your git repository using the following command:

git add . 

Commit the changes using the following command:

git commit . -m “Changes have been committed.”

Push the files to the folder you initially created using the following command:

git push -u origin master