3.5 Handle OAuth Authorization in REST API Using REST Assured



This section will guide you to understand:

* How to handle OAuth Authorization in REST API using REST Assured

**Development Environment:**

* Java 1.8
* Eclipse
* TestNG
* Maven

This guide has three subsections, namely:

3.5.1 Creating a Maven project in Eclipse

3.5.2 Handling OAuth Authorization in REST API using REST Assured

3.5.3 Pushing the code to GitHub repositories

**Step 3.5.1:** Creating a Maven project in Eclipse

OAuth: OAuth is a protocol that allows a user to grant limited access to their resources on one site to another, without having to expose their credentials.

* Open Eclipse.
* Click on File---> click on New--->Project.
* Select the Maven project and click on Next.
* Enter the Group id, Artifact id and click on Finish.
* Right click on Project---> src/test/java---> Package.
* Enter the package name and click Finish.
* Right click on Package---> New---> Class.
* Enter the class name and click on Finish.
* Add dependencies to pom.xml file.

**Step 3.5.2:** Executing the program for OAuth Authorization in REST API using REST Assured

* Write the program for OAuth Authorization in REST API using REST Assured and click on Save.

**package** Response;

import org.testng.annotations.Test;

import io.restassured.RestAssured;

import io.restassured.response.Response;

public **class** Oauth {

@Test

public void OAuth()

{

Response resp=RestAssured.given()

.auth()

.oauth2("a2c46473d65826bb118e5ae7e260d4cf604c8e982")

.post("http://192.168.1.207:8080/api/employee/search/1597534560");

System.out.println("code" +resp.getStatusCode());

System.out.println("code" +resp.getBody().asString());

}

}

* Click on run and check the output in TestNG.

[RemoteTestNG] detected TestNG version 6.14.3

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

PASSED: OAuth

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

Default test

Tests run: 1, Failures: 0, Skips: 0

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

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

Default suite

Total tests run: 1, Failures: 0, Skips: 0

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

**Step 3.5.3**: 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