Source Code:

Githublink-https://github.com/Utkarsh9991/Testing.git

**REST Assured**

**package** courseendproject1;

**import** java.io.File;

**import** org.apache.logging.log4j.LogManager; **import** org.apache.logging.log4j.Logger; **import** org.hamcrest.Matchers;

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

**import** io.restassured.RestAssured;

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

**public class** Assignment001 {

Logger logger = LogManager.*getLogger*(Assignment001.**class**); @Test(priority='1')

**public void** assignment001Post()

{

logger.info("Course End project - Assignment001 - POST request"); File file = **new** File("resources/data.json");

**int** id = RestAssured.*given*()

.baseUri("https://petstore.swagger.io/v2/pet")

.contentType(ContentType.***JSON***)

.body(file)

.when().post()

.then()

.statusCode(200)

.log().all()

.body("name", Matchers.*equalTo*("duck")).extract().path("id"); logger.trace("The status code is checked");

System.***out***.println(id);

logger.trace("Id has been captured and validated");

}

//@Test(priority='2',dependsOnMethods="assignment001Post")

**public void** assignment001Get()

{

logger.info("Course End project - Assignment001 - GET request");

**int** id = RestAssured.*given*()

.baseUri("https://petstore.swagger.io/v2/pet/987")

.when()

.get()

.then()

.statusCode(200)

.log().all()

.body("status", Matchers.*equalTo*("available")).extract().path("category.id");

System.***out***.println(id); // 0

logger.trace("Id and status has been captured and validated");

}

//@Test(priority='3',dependsOnMethods="assignment001Get")

**public void** assignment001Delete()

{

logger.info("Course End project - Assignment001 - Delete request"); RestAssured.*given*()

.baseUri("https://petstore.swagger.io/v2/pet/987")

.when()

.delete()

.then()

.statusCode(200)

.log().all()

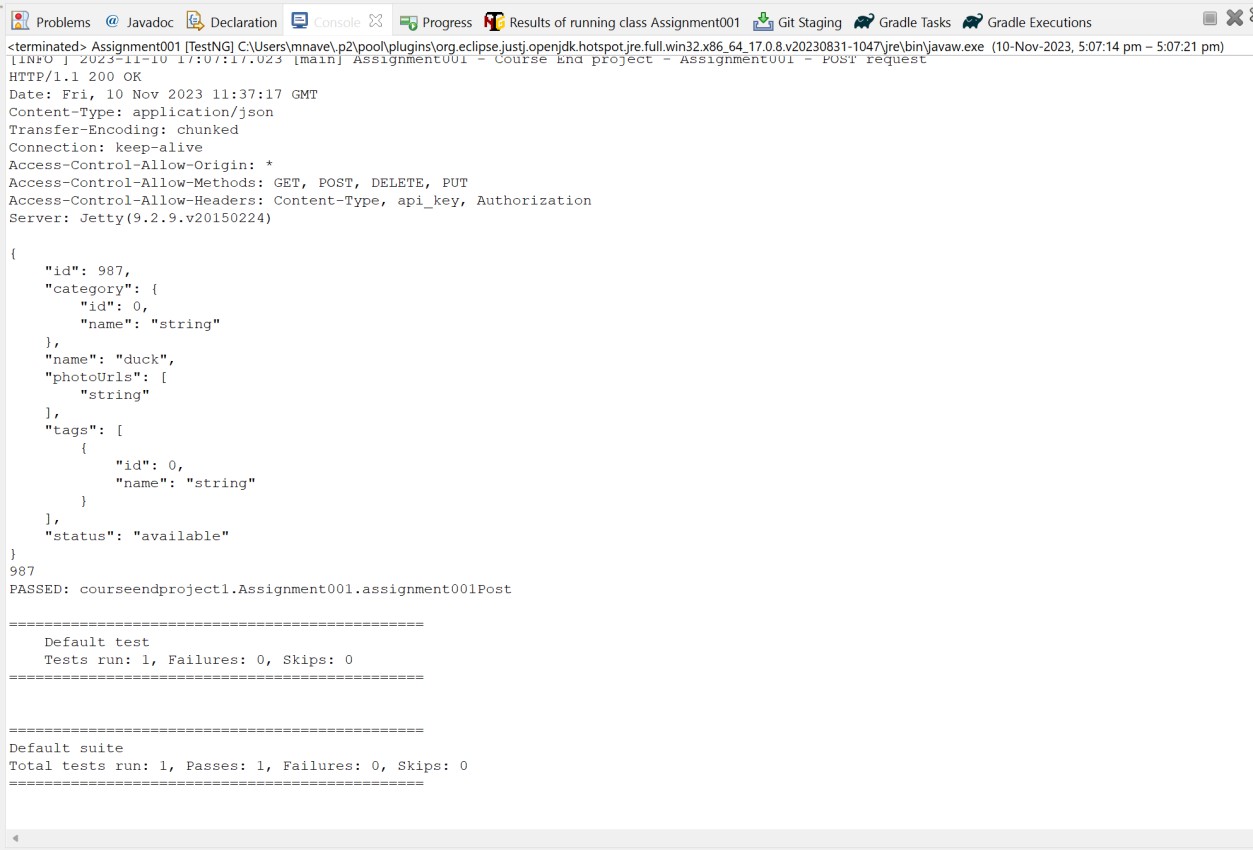
.body("code", Matchers.*equalTo*(200))

.body("type", Matchers.*equalTo*("unknown"))

.body("message", Matchers.*equalTo*("987"));

}

}



package courseendproject1; import java.io.File;

import org.apache.logging.log4j.LogManager; import org.apache.logging.log4j.Logger; import org.hamcrest.Matchers;

import org.testng.annotations.Test;

import io.restassured.RestAssured; import io.restassured.http.ContentType;

public class Assignment002 {

Logger logger = LogManager.getLogger(Assignment002.class); @Test(priority='1')

public void assignment002Post()

{

logger.info("Course End project - Assignment002 - POST request"); File file = new File("resources/data.json");

int id = RestAssured.given()

.baseUri("https://petstore.swagger.io/v2/pet")

.contentType(ContentType.JSON)

.body(file)

.when().post()

.then()

.statusCode(200)

.log().all()

.body("name", Matchers.equalTo("duck")).extract().path("id"); logger.trace("The status code is checked");

System.out.println(id);

logger.trace("Id has been captured and validated");

}

@Test(priority='2',dependsOnMethods = "assignment002Post") public void assignment002Put()

{

File file = new File("resources/dataput.json"); int id = RestAssured.given()

.baseUri("https://petstore.swagger.io/v2/pet/")

.contentType(ContentType.JSON)

.body(file)

.when()

.post()

.then()

.statusCode(200)

.log().all()

.body("status", Matchers.equalTo("available\_QA")).extract().path("id");

System.out.println(id);

}

}



# Assignment 003:

**package** courseendproject1;

**import** org.hamcrest.Matchers;

**import** org.testng.annotations.Test; **import** io.restassured.RestAssured; **public class** Assignment003demo {

@Test(priority='1')

**public void** assignment003User()

{

RestAssured.*given*()

.baseUri("https://petstore.swagger.io/v2/user/Uname001")

.when()

.get()

.then()

.statusCode(200)

.log().all()

.body("username", Matchers.*equalTo*("Uname001"))

.body("email", Matchers.*equalTo*(["Positive@Attitude.com"](mailto:Positive@Attitude.com)))

.body("userStatus", Matchers.*equalTo*(1));

}

@Test(priority='2')

**public void** assignment004login()

{

RestAssured.*given*()

.baseUri("https://petstore.swagger.io/v2/user/login")

.auth().preemptive().basic("Uname001", "@tt!tude")

.when()

.get()

.then()

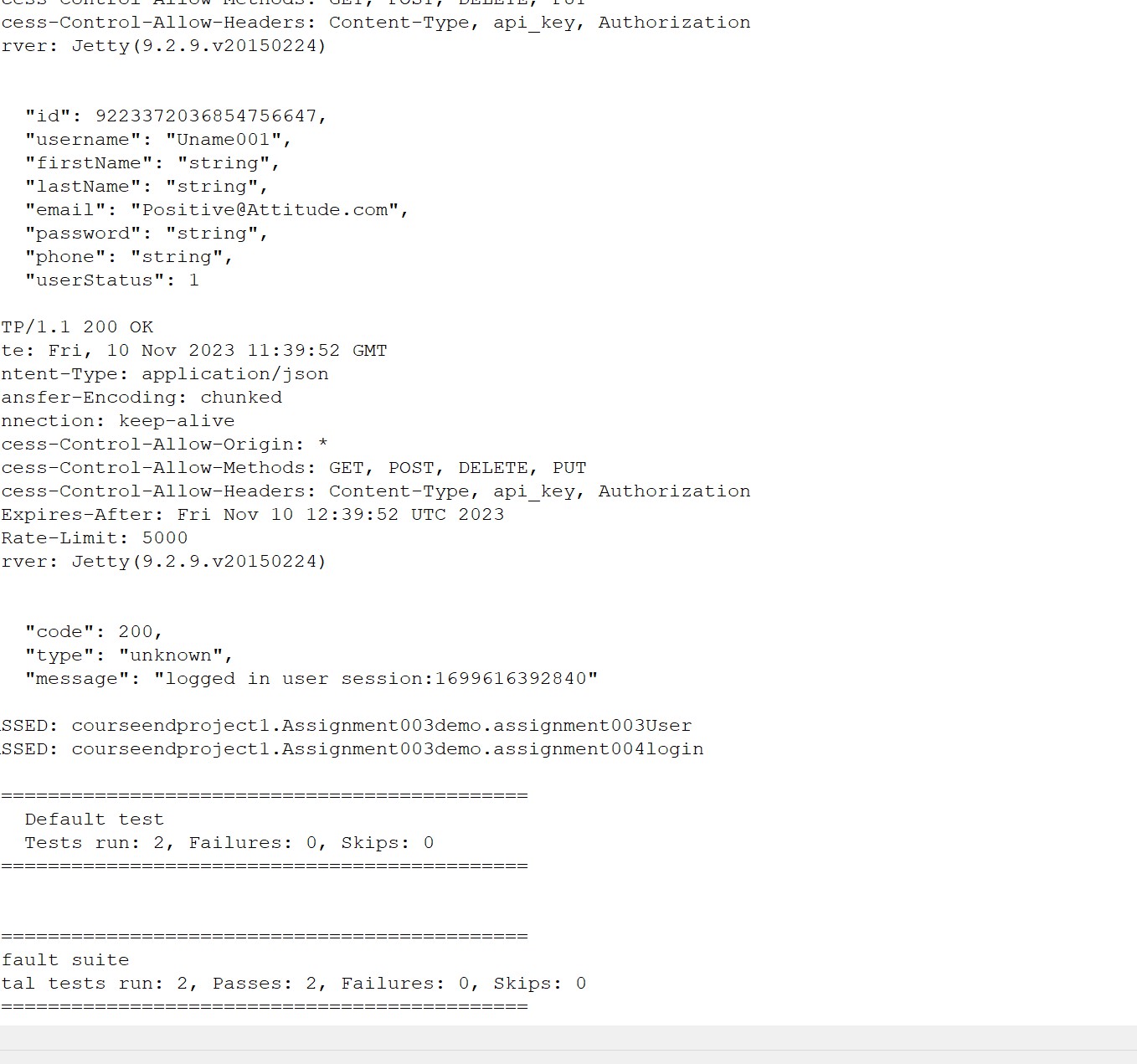
.statusCode(200)

.log().all()

.body("message", Matchers.*anything*());

}

}



**package** courseendproject1;

**import** org.hamcrest.Matchers;

**import** org.testng.annotations.Test; **import** io.restassured.RestAssured; **public class** Assignment005and006 {

@Test(priority='1')

**public void** assignment005FindByStatus()

{

RestAssured.*given*()

.baseUri("https://petstore.swagger.io/v2/pet/findByStatus")

//.queryParam("status", "available")

//.queryParam("status", "pending")

.queryParam("status", "sold")

.when()

.get()

.then()

.statusCode(200)

.log().all()

;

}

@Test(priority='2')

**public void** assignment006Logout()

{

RestAssured.*given*()

.baseUri("https://petstore.swagger.io/v2/user/logout")

.when()

.get()

.then()

.statusCode(200)

.log().all()

.body("code", Matchers.*equalTo*(200))

.body("type", Matchers.*equalTo*("unknown"))

.body("message", Matchers.*equalTo*("ok"))

;

}

}



**package** courseendproject1;

**import** org.apache.logging.log4j.LogManager; **import** org.apache.logging.log4j.Logger; **import** org.testng.annotations.Test;

**public class** Log4jDemo {

Logger logger = LogManager.*getLogger*(Log4jDemo.**class**); @Test

**public void** method1()

{

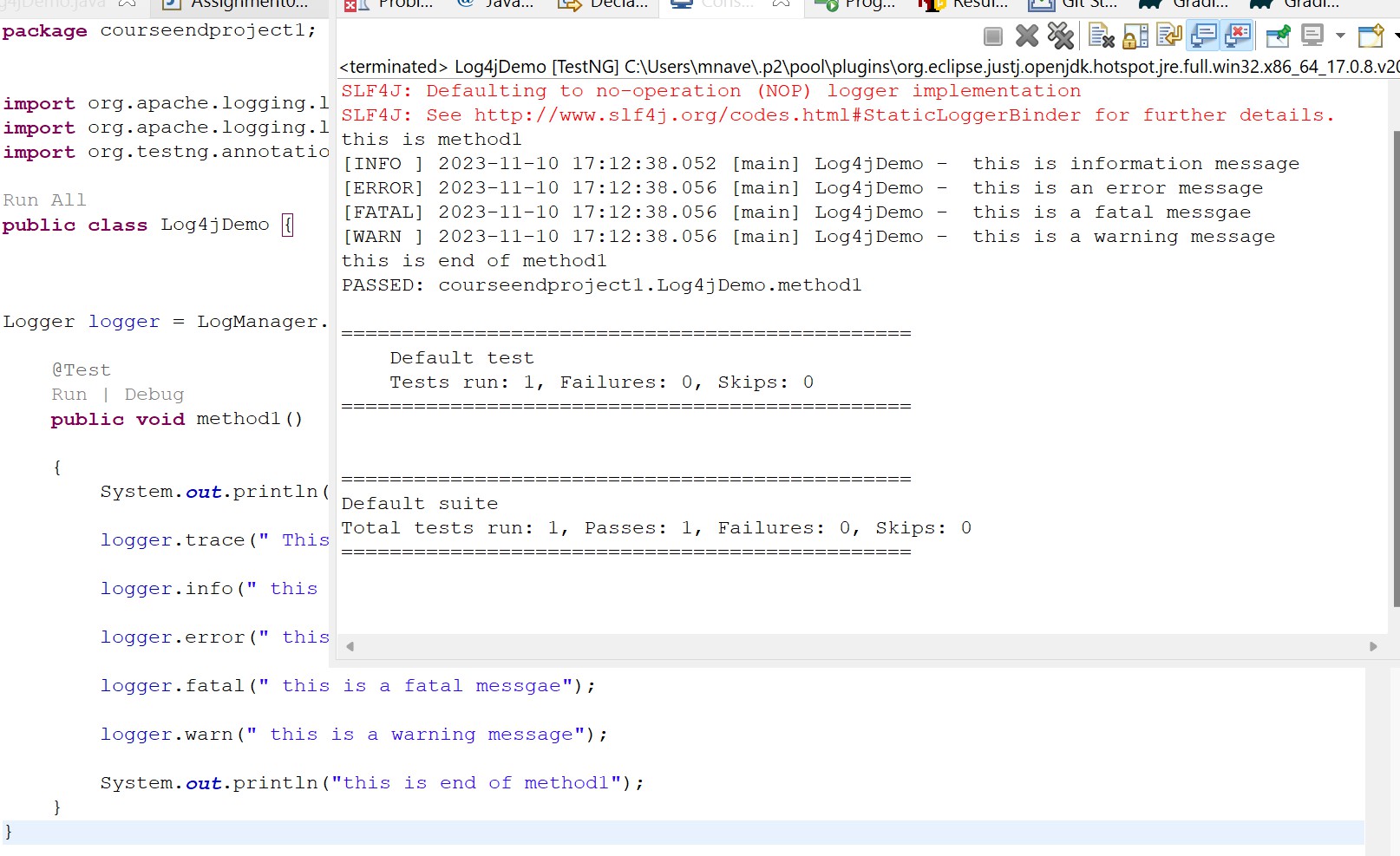
System.***out***.println("this is method1");

logger.trace(" This is trace method"); logger.info(" this is information message"); logger.error(" this is an error message"); logger.fatal(" this is a fatal messgae"); logger.warn(" this is a warning message");

System.***out***.println("this is end of method1");

}

}



Pom File for above source code:

<?xml version=*"1.0"* encoding=*"UTF-8"*?>

<project xmlns=[*"http://maven.apache.org/POM/4.0.0"*](http://maven.apache.org/POM/4.0.0)xmlns:xsi=[*"http://www.w3.org/2001/XMLSchema*](http://www.w3.org/2001/XMLSchema-instance)*-*[*instance"*](http://www.w3.org/2001/XMLSchema-instance)

xsi:schemaLocation=[*"http://maven.apache.org/POM/4.0.0*](http://maven.apache.org/POM/4.0.0)[*http://maven.apache.org/xsd/maven-4.0.0.xsd"*](http://maven.apache.org/xsd/maven-4.0.0.xsd)>

<modelVersion>4.0.0</modelVersion>

<groupId>ATE-phase3-RestAssured</groupId>

<artifactId>ATE-phase3-RestAssured</artifactId>

<version>0.0.1-SNAPSHOT</version>

<name>ATE-phase3-RestAssured</name>

<!-- FIXME change it to the project's website -->

<url>[http://www.example.com](http://www.example.com/)</url>

<properties>

<project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>

<maven.compiler.source>1.7</maven.compiler.source>

<maven.compiler.target>1.7</maven.compiler.target>

</properties>

<dependencies>

<dependency>

<groupId>junit</groupId>

<artifactId>junit</artifactId>

<version>4.11</version>

<scope>test</scope>

</dependency>

<!-- https://mvnrepository.com/artifact/io.cucumber/cucumber-java -->

<dependency>

<groupId>io.cucumber</groupId>

<artifactId>cucumber-java</artifactId>

<version>7.10.1</version>

</dependency>

<!-- https://mvnrepository.com/artifact/io.cucumber/cucumber-java -->

<dependency>

<groupId>io.cucumber</groupId>

<artifactId>cucumber-junit</artifactId>

<version>7.10.1</version>

<scope>test</scope>

</dependency>

<!-- https://mvnrepository.com/artifact/io.cucumber/cucumber-java -->

<dependency>

<groupId>io.cucumber</groupId>

<artifactId>cucumber-core</artifactId>

<version>7.10.1</version>

</dependency>

<!-- https://mvnrepository.com/artifact/io.rest-assured/rest-assured -->

<dependency>

<groupId>io.rest-assured</groupId>

<artifactId>rest-assured</artifactId>

<version>5.3.0</version>

<scope>test</scope>

</dependency>

<!-- https://mvnrepository.com/artifact/io.rest-assured/rest-assured-common -->

<dependency>

<groupId>io.rest-assured</groupId>

<artifactId>rest-assured-common</artifactId>

<version>5.3.0</version>

</dependency>

<!-- https://mvnrepository.com/artifact/io.rest-assured/rest-assured-all -->

<dependency>

<groupId>io.rest-assured</groupId>

<artifactId>rest-assured-all</artifactId>

<version>5.3.0</version>

<scope>test</scope>

</dependency>

<!-- https://mvnrepository.com/artifact/io.rest-assured/json-path -->

<dependency>

<groupId>io.rest-assured</groupId>

<artifactId>json-path</artifactId>

<version>5.3.0</version>

<scope>test</scope>

</dependency>

<!-- https://mvnrepository.com/artifact/org.json/json -->

<dependency>

<groupId>org.json</groupId>

<artifactId>json</artifactId>

<version>20180813</version>

</dependency>

<!-- https://mvnrepository.com/artifact/org.hamcrest/hamcrest -->

<dependency>

<groupId>org.hamcrest</groupId>

<artifactId>hamcrest</artifactId>

<version>2.2</version>

<scope>test</scope>

</dependency>

<!-- https://mvnrepository.com/artifact/com.fasterxml.jackson.core/jackson-databind

-->

<dependency>

<groupId>com.fasterxml.jackson.core</groupId>

<artifactId>jackson-databind</artifactId>

<version>2.15.2</version>

</dependency>

<!-- https://mvnrepository.com/artifact/org.testng/testng -->

<dependency>

<groupId>org.testng</groupId>

<artifactId>testng</artifactId>

<version>7.7.1</version>

<scope>test</scope>

</dependency>

<dependency>

<groupId>org.apache.logging.log4j</groupId>

<artifactId>log4j-api</artifactId>

<version>2.14.1</version>

</dependency>

<!-- https://mvnrepository.com/artifact/org.apache.logging.log4j/log4j-core -->

<dependency>

<groupId>org.apache.logging.log4j</groupId>

<artifactId>log4j-core</artifactId>

<version>2.14.1</version>

</dependency>

</dependencies>

<build>

<pluginManagement><!-- lock down plugins versions to avoid using Maven defaults (may be moved to parent pom) -->

<plugins>

<!-- clean lifecycle, see https://maven.apache.org/ref/current/maven- core/lifecycles.html#clean\_Lifecycle -->

<plugin>

<artifactId>maven-clean-plugin</artifactId>

<version>3.1.0</version>

</plugin>

<!-- default lifecycle, jar packaging: see https://maven.apache.org/ref/current/maven-core/default- bindings.html#Plugin\_bindings\_for\_jar\_packaging -->

<plugin>

<artifactId>maven-resources-plugin</artifactId>

<version>3.0.2</version>

</plugin>

<plugin>

<artifactId>maven-compiler-plugin</artifactId>

<version>3.8.0</version>

</plugin>

<plugin>

<artifactId>maven-surefire-plugin</artifactId>

<version>2.22.1</version>

</plugin>

<plugin>

<artifactId>maven-jar-plugin</artifactId>

<version>3.0.2</version>

</plugin>

<plugin>

<artifactId>maven-install-plugin</artifactId>

<version>2.5.2</version>

</plugin>

<plugin>

<artifactId>maven-deploy-plugin</artifactId>

<version>2.8.2</version>

</plugin>

<!-- site lifecycle, see https://maven.apache.org/ref/current/maven- core/lifecycles.html#site\_Lifecycle -->

<plugin>

<artifactId>maven-site-plugin</artifactId>

<version>3.7.1</version>

</plugin>

<plugin>

<artifactId>maven-project-info-reports-plugin</artifactId>

<version>3.0.0</version>

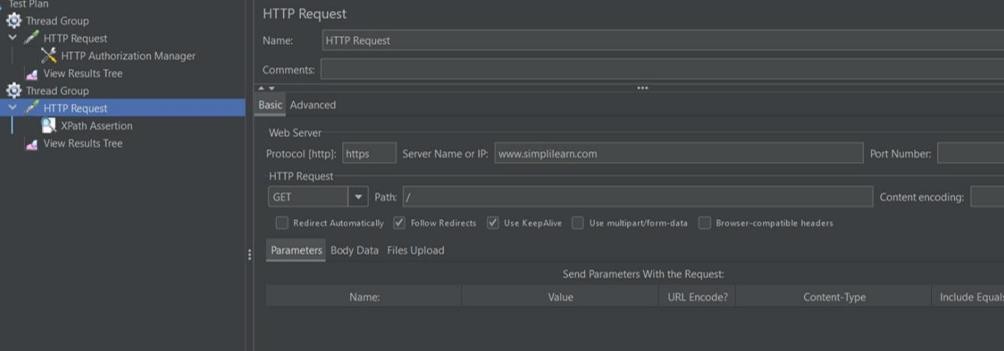
</plugin>

</plugins>

</pluginManagement>

</build>

</project>



OutPut:

Scenario: Rest API testing on reqres.in # src/test/java/features/lessonend.feature:4

Given User send a Post request to create a user and validates status # steps.LessonEndProject.user\_send\_a\_post\_request\_to\_create\_a\_user\_and\_validates\_status()

Given User sends a Get request to get a user and validates status # steps.LessonEndProject.user\_sends\_a\_get\_request\_to\_get\_a\_user\_and\_validates\_status()

HTTP/1.1 200 OK

Date: Fri, 19 Dec 2023 11:20:31 GMT

Content-Type: application/json; charset=utf-8

Transfer-Encoding: chunked

Connection: keep-alive

Report-To: {"group":"heroku-nel","max\_age":3600,"endpoints":[{"url":"https://nel.heroku.com/reports?ts=1699509212&sid=c4c9725f-1ab0-44d8-820f-430df2718e11&s=ESfqs4kibMK2oKXzrt5OBvS3cr7SWb9h%2Fv0Ve69kcoA%3D"}]}

Reporting-Endpoints: heroku-nel=https://nel.heroku.com/reports?ts=1699509212&sid=c4c9725f-1ab0-44d8-820f-430df2718e11&s=ESfqs4kibMK2oKXzrt5OBvS3cr7SWb9h%2Fv0Ve69kcoA%3D

Nel: {"report\_to":"heroku-nel","max\_age":3600,"success\_fraction":0.005,"failure\_fraction":0.05,"response\_headers":["Via"]}

X-Powered-By: Express

Access-Control-Allow-Origin: \*

Etag: W/"2c1-N6Rqerxquq2kgQhL51EiSg4x0R8"

Via: 1.1 vegur

Cache-Control: max-age=14400

CF-Cache-Status: HIT

Age: 2267

Vary: Accept-Encoding

Server: cloudflare

CF-RAY: 823de81e1be2795c-AMD

Content-Encoding: gzip

{

"page": 1,

"per\_page": 6,

"total": 12,

"total\_pages": 2,

"data": [

{

"id": 1,

"name": "cerulean",

"year": 2000,

"color": "#98B2D1",

"pantone\_value": "15-4020"

},

{

"id": 2,

"name": "fuchsia rose",

"year": 2001,

"color": "#C74375",

"pantone\_value": "17-2031"

},

{

"id": 3,

"name": "true red",

"year": 2002,

"color": "#BF1932",

"pantone\_value": "19-1664"

},

{

"id": 4,

"name": "aqua sky",

"year": 2003,

"color": "#7BC4C4",

"pantone\_value": "14-4811"

},

{

"id": 5,

"name": "tigerlily",

"year": 2004,

"color": "#E2583E",

"pantone\_value": "17-1456"

},

{

"id": 6,

"name": "blue turquoise",

"year": 2005,

"color": "#53B0AE",

"pantone\_value": "15-5217"

}

],

"support": {

"url": "https://reqres.in/#support-heading",

"text": "To keep ReqRes free, contributions towards server costs are appreciated!"

}

}

Given User sends a get request to get list of users and validates status # steps.LessonEndProject.user\_sends\_a\_get\_request\_to\_get\_list\_of\_users\_and\_validates\_status()

1 Scenarios (1 passed)

3 Steps (3 passed)

0m3.599s