SourceCode:

1.RetrieveAllProducts:

```
package com.simplilearn.restassured;


import static org.hamcrest.CoreMatchers.equalTo;


import org.apache.log4j.Logger;
import org.testng.annotations.Test;


import io.restassured.RestAssured;


public class RetrievetAllProducts {


	private static final String BASE_URL = "http://localhost:9010";
	private static final Logger logger = Logger.getLogger(RetrievetAllProducts.class);
	String response=null;


	@Test(description= "Test to Get all product from the store")
	public void testGetAllProducts() {
		try {
			logger.info("Start :: Test to Get all products from the store");
			RestAssured.given().baseUri(BASE_URL)
			.when().get("/get-shoes").then().statusCode(200).and()
			.body("message", equalTo("13 Shoes Fetched Successfully."));



			response = RestAssured.given().baseUri(BASE_URL)
			.when().get("/get-shoes").getBody().asString();
		}catch (Exception e) {
```

```
            logger.error("Exception object :: " +e.toString());
            logger.error("End Exception :: " +e.getLocalizedMessage());
        }
        logger.info("Response ::" +response);
        logger.info("End  :: Test to Get all products from the store");
    }
}
```

2.RetrieveAllRegisteredUsers:

```
package com.simplilearn.restassured;

import static org.hamcrest.CoreMatchers.equalTo;

import org.apache.log4j.Logger;
import org.testng.annotations.Test;

import io.restassured.RestAssured;

public class RegisteredUsers {


    private static final String BASE_URL = "ttp://localhost:9010";
    private static final Logger logger = Logger.getLogger(RegisteredUsers.class);
    String response=null;

    @Test(description= "Test to Get all Registered Users")
    public void testGetAllRegisteredUsers() {
        try {
            logger.info("Start :: Test to Get all products from the store");
            RestAssured.given().baseUri(BASE_URL)
            .when().get("/get-users").then().statusCode(200).and()
            .body("message", equalTo("3 Users Fetched Successfully."));



            response = RestAssured.given().baseUri(BASE_URL)
            .when().get("/get-users").getBody().asString();
        }catch (Exception e) {
            logger.error("Exception object :: " +e.toString());
            logger.error("End Exception :: " +e.getLocalizedMessage());
        }
        logger.info("Response ::" +response);
        logger.info("End  :: Test to Get all Registered Users");
    }
}
```

3.AddTheProduct:

```
package com.simplilearn.restassured;

import static org.hamcrest.CoreMatchers.equalTo;
import org.apache.log4j.Logger;
import org.testng.annotations.Test;
import io.restassured.RestAssured;
import io.restassured.http.ContentType;

public class AddTheProduct {

    private static String BASE_URL = "http://localhost:9010";
      private static Logger logger = Logger.getLogger(AddTheProduct.class);
      String response = null;
```

```
@Test(description = "Test to create a product")
public void testAddProduct() {
   try {
      logger.info("Start :: Test to create a product ");
      int id = 101;
      String image_url = "image_url";
      String name = "sampleShoe";
      String category = "Runnings";
      int sizes = 9;
      int price = 1000;

      RestAssured.given().baseUri(BASE_URL)
            .queryParam("id",id).queryParam("image",image_url)
            .queryParam("name",name).queryParam("category",category)
            .queryParam("sizes",sizes).queryParam("price",price)
            .when().post("/add-shoe")
            .then().assertThat().statusCode(200)
            .body("message",equalTo("sampleShoe Added Successfully."));

      response = RestAssured.given().baseUri(BASE_URL)
                 .queryParam("id",id).queryParam("image",image_url)
            .queryParam("name",name).queryParam("category",category)
            .queryParam("sizes",sizes).queryParam("price",price)
            .when().post("/add-shoe")
            .getBody().asString();
   } catch (Exception e) {
      logger.error("Exception Object :: " + e.toString());
      logger.error("End Exception :: " + e.getLocalizedMessage());
   }
```

```
             logger.info("Response :: " + response);
             logger.info("End :: Test to create a product");
           }

}
```

4.UpdateThePRoduct:

```
package com.simplilearn.restassured;

import static org.hamcrest.CoreMatchers.equalTo;

import org.apache.log4j.Logger;
import org.testng.annotations.Test;

import io.restassured.RestAssured;

public class UpdateProduct {



        private static final String BASE_URL="http://localhost:9010";
        private static Logger logger=Logger.getLogger(UpdateProduct.class);
        String response=null;
        @Test(description="Test to update the product in the store")
        public void testGetProduct() {
                try {
                        int id = 101;
       String image= "update image";
       String name = "updated Shoe";
       String category = " updated Runnings";
```

```
        int sizes = 12;
        int price = 1099;
                logger.info("Start :: Test to update the product in the store");


                RestAssured.given().baseUri(BASE_URL)
                .queryParam("id",id).queryParam("image",image)
    .queryParam("name",name).queryParam("category",category)
    .queryParam("sizes",sizes).queryParam("price",price)
                .when().put("/update-shoe").then()
                .assertThat().statusCode(200)
                .body("message", equalTo("updated Shoe Updated Successfully."));



                response = RestAssured.given().baseUri(BASE_URL)
                            .queryParam("id",id).queryParam("image",image)
                    .queryParam("name",name).queryParam("category",category)
                    .queryParam("sizes",sizes).queryParam("price",price)
                    .when().put("/update-shoe").getBody().asString();
        }catch(Exception e) {
                logger.error("Exception Object :: "+e.toString());
                logger.error("End Exception :: "+e.getLocalizedMessage());
        }
                logger.info("Reponse :: "+response);
                logger.info("End :: Test to update the product in the store");
        }


}
```

5.DeleteTheProduct:

```
package com.simplilearn.restassured;
```

```
import org.apache.log4j.Logger;
import org.testng.annotations.Test;
import static org.hamcrest.CoreMatchers.equalTo;
import io.restassured.RestAssured;

public class DeleteProduct {

    private static final String BASE_URL = "ttp://localhost:9010";
    private static final Logger logger = Logger.getLogger(DeleteProduct.class);
    String response=null;

    @Test(description= "Test to delete product from the store")
    public void testDeleteProduct() {
        try {
            logger.info("Start :: Test to delete the product from the store");
            int id = 101;
            RestAssured.given().baseUri(BASE_URL)
            .when().delete("/delete-shoe?id="+id).then().statusCode(200).and()
            .body("message", equalTo("Shoe with ID 101 Deleted Successfully."));


            response = RestAssured.given().baseUri(BASE_URL)
            .when().delete("/delete-shoe?id="+id).getBody().asString();
        }catch (Exception e) {
            logger.error("Exception object :: " +e.toString());
            logger.error("End Exception :: " +e.getLocalizedMessage());
        }
        logger.info("Response ::" +response);
```

```
                logger.info("End  :: Test to delete product from the store");
        }


}
```