

Source Code:

StepDefinitions:

DeleteProductsSteps:

```
package com.cimplilearn.cucumber.stepdefinitions;

import static org.hamcrest.CoreMatchers.containsString;
import io.cucumber.java.en.Given;
import io.cucumber.java.en.Then;
import io.cucumber.java.en.When;
import io.restassured.RestAssured;
import io.restassured.response.Response;
import io.restassured.response.ValidatableResponse;
import io.restassured.specification.RequestSpecification;

public class DeleteProductSteps {

    private Response response;
    private ValidatableResponse json;
    private RequestSpecification request;

    private String BASE_URL = "http://localhost:9010";

    @Given("there is an existing shoe with ID {int}")
    public void there_is_an_existing_shoe_with_id(Integer int1) {
        request = RestAssured.given().baseUrl(BASE_URL);
    }

    @When("hit the DELETE request for the shoe with ID {int}")
    public void hit_the_delete_request_for_the_shoe_with_id(Integer shoeld) {
        response = request.delete("/delete-shoe?id=" + shoeld);
    }
}
```

```
}
```

```
@Then("the response code should be {int}")
```

```
public void the_response_code_should_be(Integer code) {
```

```
    json = response.then().statusCode(code);
```

```
}
```

```
@Then("the response should indicate successful deletion")
```

```
public void the_response_should_indicate_successful_deletion() {
```

```
    response.then().body("message", containsString("Shoe with ID 101 Deleted  
Successfully"));
```

```
}
```

```
}
```

GetRequestSteps:

```
package com.cimplilearn.cucumber.stepdefinitions;
```

```
import org.hamcrest.Matchers;
```

```
import io.cucumber.java.en.Given;
```

```
import io.cucumber.java.en.Then;
```

```
import io.cucumber.java.en.When;
```

```
import io.restassured.RestAssured;
```

```
import io.restassured.response.Response;
```

```
import io.restassured.response.ValidatableResponse;
```

```
import io.restassured.specification.RequestSpecification;
```

```
public class GetRequestSteps {
```

```

private Response response;

private ValidatableResponse json;

private RequestSpecification request;


private String BASE_URL = "http://localhost:9010";


@Given("I am interested in viewing the available shoes")
public void i_am_interested_in_viewing_the_available_shoes() {
    request=RestAssured.given().baseUrl(BASE_URL);
}


@When("I send a request to retrieve the list of products")
public void i_send_a_request_to_retrieve_the_list_of_products() {
    response=request.when().get("/get-shoes");
}


@Then("I should receive a successful response with status code {int}")
public void i_should_receive_a_successful_response_with_status_code(Integer
code) {
    json=response.then().statusCode(code);
    System.out.println(json.toString());
}


@Then("the response should contain a valid list of sports shoes")
public void the_response_should_contain_a_valid_list_of_sports_shoes() {
    response.then().body("shoes", Matchers.notNullValue());
}

}

```

3.GetUserSteps:

```
package com.cimplilearn.cucumber.stepdefinitions;
```

```
import org.hamcrest.Matchers;
```

```
import io.cucumber.java.en.Given;
```

```
import io.cucumber.java.en.Then;
```

```
import io.cucumber.java.en.When;
```

```
import io.restassured.RestAssured;
```

```
import io.restassured.response.Response;
```

```
import io.restassured.response.ValidatableResponse;
```

```
import io.restassured.specification.RequestSpecification;
```

```
public class GetUserSteps {
```

```
    private Response response;
```

```
    private ValidatableResponse json;
```

```
    private RequestSpecification request;
```

```
    private String BASE_URL = "http://localhost:9010";
```

```
    @Given("I am interested in viewing the registered users")
```

```
    public void i_am_interested_in_viewing_the_registered_users() {
```

```
        request=RestAssured.given().baseUrl(BASE_URL);
```

```
    }
```

```
    @When("I send a request to retrieve the users")
```

```
    public void i_send_a_request_to_retrieve_the_users() {
```

```
        response=request.when().get("/get-users");
```

```
}
```

```
    @Then("I want to receive a successful response with status code {int}")
    public void i_want_to_receive_a_successful_response_with_status_code(Integer
code) {
        json=response.then().statusCode(code);
        System.out.println(json.toString());
    }
```

```
    @Then("the response should contain a valid list of users")
    public void the_response_should_contain_a_valid_list_of_users() {
        response.then().body("users", Matchers.notNullValue());
    }
}
```

4.PostRequestSteps:

```
package com.cimplilearn.cucumber.stepdefinitions;

import io.cucumber.java.en.Given;
import io.cucumber.java.en.Then;
import io.cucumber.java.en.When;
import io.restassured.RestAssured;
import io.restassured.response.Response;
import io.restassured.response.ValidatableResponse;
import io.restassured.specification.RequestSpecification;

public class PostRequestSteps {
```

```
private Response response;  
private ValidatableResponse json;  
private RequestSpecification request;  
private String requestBody;
```

```
private String BASE_URL = "http://localhost:9010";
```

```
@Given("Open the API base URL")
```

```
public void open_the_api_base_url() {  
    request=RestAssured.given().baseUrl(BASE_URL);  
}
```

```
@Given("Enter the shoe details")
```

```
public void enter_the_shoe_details() {  
    requestBody = "id=" + 101 + "&image=image_url&name=Sample  
Shoes&category=Sports Shoe&sizes=9&price=1999";  
}
```

```
@When("Hit the POST request to create a new shoe")
```

```
public void hit_the_post_request_to_create_a_new_shoe() {  
    int id = 101;  
    String image_url = "image_url";  
    String name = "sampleShoe";  
    String category = "Runnings";  
    int sizes = 9;  
    int price = 1000;  
    response=request.queryParam("id",id).queryParam("image",image_url)  
        .queryParam("name",name).queryParam("category",category)  
        .queryParam("sizes",sizes).queryParam("price",price).when().post("/add-shoe");
```

```
}
```

```
@Then("The response code should be {int}")
```

```
public void the_response_code_should_be(Integer code) {
```

```
    json=response.then().statusCode(code);
```

```
    System.out.println(json);
```

```
}
```

```
@Then("Print the result of adding new shoe")
```

```
public void print_the_result_of_adding_new_shoe() {
```

```
    System.out.println(response.getBody().asString());
```

```
}
```

```
}
```

5.UpdateShoesRequest:

```
package com.cimplilearn.cucumber.stepdefinitions;
```

```
import io.cucumber.java.en.Given;
```

```
import io.cucumber.java.en.Then;
```

```
import io.cucumber.java.en.When;
```

```
import io.restassured.RestAssured;
```

```
import io.restassured.response.Response;
```

```
import io.restassured.response.ValidatableResponse;
```

```
import io.restassured.specification.RequestSpecification;
```

```
public class UpdateshoeSteps {
```

```
    private Response response;
```

```
    private ValidatableResponse json;
```

```
private RequestSpecification request;
```

```
private String BASE_URL = "http://localhost:9010";
```

```
@Given("open the update product base url")
```

```
public void open_the_update_product_base_url() {
```

```
    request=RestAssured.given().baseUrl(BASE_URL);
```

```
}
```

```
@When("hit the PUT request to update the shoe with ID {int}")
```

```
public void hit_the_put_request_to_update_the_shoe_with_id(Integer shoeld) {
```

```
    int id = shoeld;
```

```
String image= "update image";
```

```
String name = "updated Shoe";
```

```
String category = " updated Runnings";
```

```
int sizes = 12;
```

```
int price = 1099;
```

```
    response =request
```

```
        .queryParams("id",id).queryParams("image",image)
```

```
        .queryParams("name",name).queryParams("category",category)
```

```
        .queryParams("sizes",sizes).queryParams("price",price)
```

```
        .when().put( "/update-shoe?id=" );
```

```
    }
```

```
@Then("the response code should be equal to {int}")
```

```
public void the_response_code_should_be_equal_to(Integer expectedStatusCode)
```

```
{
```

```
    json=response.then().statusCode(expectedStatusCode);
```



```
}
```

```
@Then("the response should indicate successful update")  
public void the_response_should_indicate_successful_update() {  
    System.out.println(json.toString());  
}
```

```
}
```

TestRunner:

```
package com.simplilearn.cucumber.testrunner;
```

```
import org.junit.runner.RunWith;
```

```
import io.cucumber.junit.Cucumber;
```

```
import io.cucumber.junit.CucumberOptions;
```

```
@RunWith(Cucumber.class)
```

```
@CucumberOptions(features = "src/test/resources/Features", glue =  
{"com/simplilearn/cucumber/stepdefinitions" })
```

```
public class TestRunner {
```

```
}
```