**Test Summary Report**

* Initially started with creating a Maven Project in IDE
* Added required dependencies in POM.xml file such as io.rest-assured, org.testng, etc. to execute the test cases.
* Here we are taking the use of RestAssured framework with BDD style in order to execute the test cases.
* Imported required repositories through dependencies to all test cases.
* Validating Status Code and status line along with JSON response for all the test cases.
* Written Code with following method:

1. Given () - set cookies, add auth, add param, set headers info etc.
2. When () - get, post, put, delete
3. Then () - validate status code, extract response

* Created a TestNG.xml file to run all test cases at a time.
* One of the objectives is to print the same output on IDE console as what you received on the browser through Rest Assured.
* In order to view the results in allure, created dependencies for allure such as io.qameta.allure in POM.xml file.
* In order to get Allure results, need to go to right click on project > Show In > System Explorer > open project > run as “Allure Serve” in command prompt.
* Allure report displays the detailed results from the REST API Regression Testing using TestNG.
* Allure Results displays as the following:

1. POST Request Test Description: Post Method to Create a Transaction.
2. GET Request Test Description: Get Method to Retrieve the List of Transactions.
3. GET WITHSEARCHKEYWORD Request Test Description: Get Method with Search Criteria.
4. DELETE WITHSEARCHKEYWORD Request Test Description: Delete Method with Correct Id or Incorrect Id in List of Transactions.

**Steps Performed in Post Request:**

* POST method is used to send data to a server to create/update the resource.
* Initially imported all repositories.
* Created a class called “Task\_Post\_Request”.
* For post request, we are supposed to pass request body parameters along with headers.
* contentType() - Content-Type is the type of resource that used to send data.
* body() - Body is nothing but carries the details that we want to POST.
* post() - Is used to perform POST request.
* Major validation in Post will be through “201” status code.
* As a result, it should post all the created requests.

**Steps Performed in Get Request:**

* GET method is used to request data from a specific resource.
* Should get all the list of transactions to Get request.
* Validated status code and status line along with JSON line.
* For Get Request used status code “200” to validate.
* Header and Accepting Content type is JSON.
* get() - Is used to perform GET request.
* Based on the condition, whichever requests are satisfied output will be displayed accordingly.

**Steps Performed in Get WithSearchKeyword:**

* Get WithSearchKeyword method is used to search specific data from the resource by giving a keyword.
* Added few queryparameters() to validate Get request with a search keyword.
* Here we are getting list of transactions with Get WithSearchKeyword.

Upon on the keyword given, based on that it’ll validate the status code

* Once the given parameters are initialized, if there is any match on the search keyword then the output will be displayed as status code “200” Else it should display an error.

**Steps Performed in Delete WithSearchKeyword:**

* Delete WithSearchKeyword method is used to delete specific data from the resource by giving a keyword.
* Added pathparameter() to search with specific keyword.
* With the given ID, it’ll validate the test case.
* There are two criteria’s:

1. If we give Proper ID, created status code as “204” and it will delete the record.
2. If we give Incorrect ID, created status code as “404” and will display message as “Transaction not found”.

* With “log().all” we can able to print the JSON response.