Service automation using REST Assured:

To start with service automation what are the things we know it how to write to POJO classes, how to convert JSON to POJO, How to send the data to the server and how we will receive response from the server.

So Rest Assured web service testing we need few dependency rest-assured, GSON, JSON, TestNG,

Ones you have added all dependency in pom.xml then you need to right click on the pom.xml, select and click on Run As and then select maven install. And you dependencies will add to the project. after that right click on the project select and click maven and select update project then click on force update and finally click on ok.

When we say REST API automation it means every API has a payload and payload means the data which I consume. For example we say registration of student it requires certain data, this information is nothing but payload.

Similarly in service automation every service has unique address and every service consume data based on the calls so when you have POST call it requires data, when you have PUT call it requires data and finally DELETE call it also requires data, sometimes it does not requires data.

For POST call you will get POST call information from development team and it contain end point URL also you will get corresponding request data and response data.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Now we will perform operation for POST call

First I’ll copy the request data from given file then I’ll go to ***json formatter*** and you will see the data which are given by developer is in the form array so we have to use hash map concept to post data as well as go update data.

Now I’ll go with POJO classes so again I’ll convert JSON code code in to POJO class for that we have to use <http://www.jsonschema2pojo.org/> link to convert this JSON class to POJO class. And pest your request payload there. And click on radio button on java, JSON, GSON and then click on preview button and your code will be converted in to POJO class. Whenever you will click on preview button then corresponding class will get generated.

Now already I have created package folder for request POJO class so I’ll make one class called Adress or any name which is generated by POJO class name. and save it on this particular package. And other class also created by using name Example.class so you can give any name to this.

After that you one service URL for that you have to make one class called serviceULR in the service folder or package.

And go to service class create one method called createPersonAPI(). In this method you will create person based on POJO classes.

Now go to your CreatePerson POJO class and you have seen information like String name, String surname, and private Address address.

The private Address address, it means that it is address and it is return type Address and what does means that which means this address class which has link of address class. And this address has link of our Address java class. So that’s why they are like that. So this createPerson java POJO class has an object of Address POJO java class. So first what we will do here is we will go to our service java class. We will create object of person. And by using this class reference we can set the name and surname value.

***CreatePerson createperson = new CreatePerson();***

***createperson.setName(name);***

***createperson.setSurname(surname);***

Now we will create object of Address class. Ones we create the object of Address class first we will need to set address of class. Because we have one address setter of that class

***createperson.setAddress(address);***

After that using address class object we will set the data of address class.

***createperson.setAddress(address);***

***address.setCity(city);***

***address.setLandmark(landmark);***

***address.setState(state);***

***address.setZipcode(zipcode);***

this is initialization of the data.

Now we will POST the data with this information, to post the data in the server what we have to do is we are making the object of Rest Assured class. And by using this class we will invoke method called given() method.

Given() is the static method due to this reason we are able to call through the class name that is RequestSpecification class. And RequestSpecification is return class of this method.

***RequestSpecification requestSpecification = RestAssured.given();***

And this requestspecification class has different method. So first we will add header. And the header is a JSON header so we can add first content header with its value that is application/json.

***requestSpecification.contentType("application/json");***

so this is the content type with this type we will send the data to the server. Now after setting the content type we want to accept it so for that we will use accept method.

***requestSpecification.accept("application/json");***

After accepting content type we need to convert POJO object to JSON object, for this convert we will use JSON object and convert POJO class output to JSON class output. And with this JSON object we can pass the POJO class object.

***JSONObject jsonobject = new JSONObject(createperson);***

And with the reference of requestspecification we can send the body of this json object.

***requestSpecification.body(jsonobject);***

After supplying the body we can post the data with URI, this URI is nothing but your service URL.

***requestSpecification.post(ServiceURL.url);***

Whenever you will post URL it will return response.

***Response response = requestSpecification.post(ServiceURL.url);***

After all this performing we will return this response.

**return** response; and change the method type is Response.

No your one API is ready with this API we can test your create person API. And this API will give you response as JSON

To Test the service API you have to create object of service class and by using object class we can access one method that is createPerson which is you have created in the service and pass the data to this method.