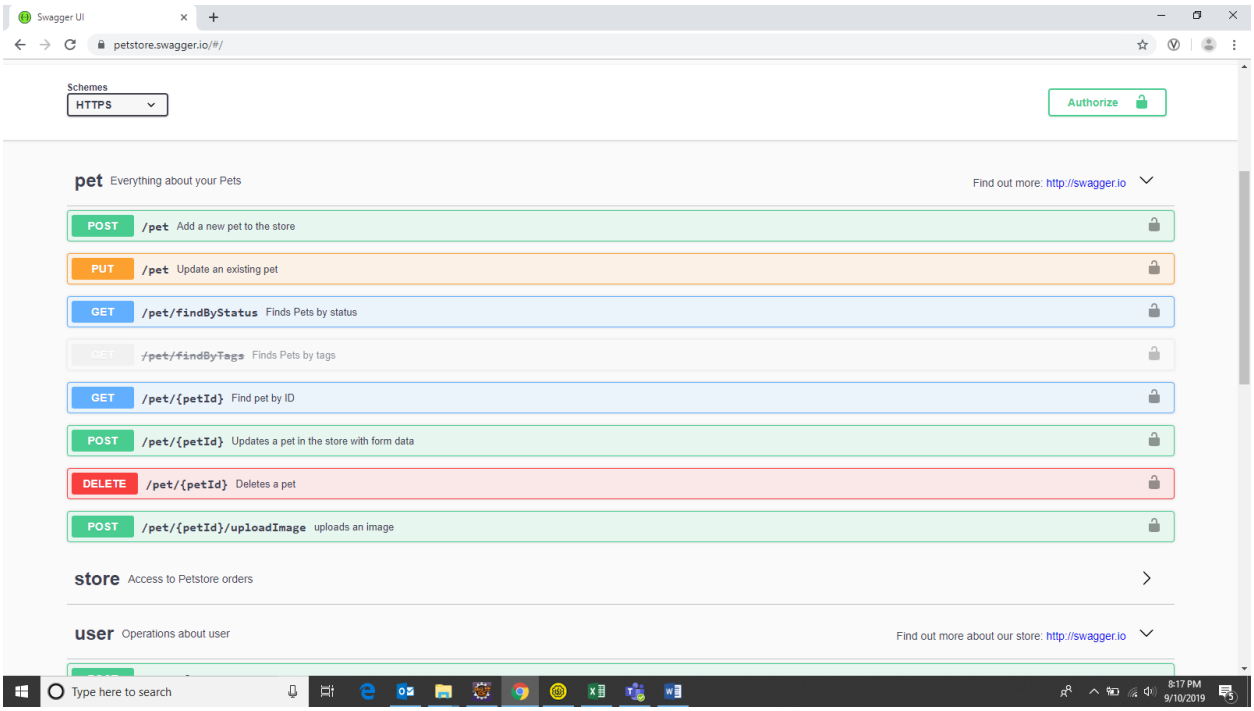
Groovy Script for REST Request

Here we are performing all the four operations (GET , PUT, POST , DELETE) in groovy script using a swagger link:- https://petstore.swagger.io/#/

(GET Method)

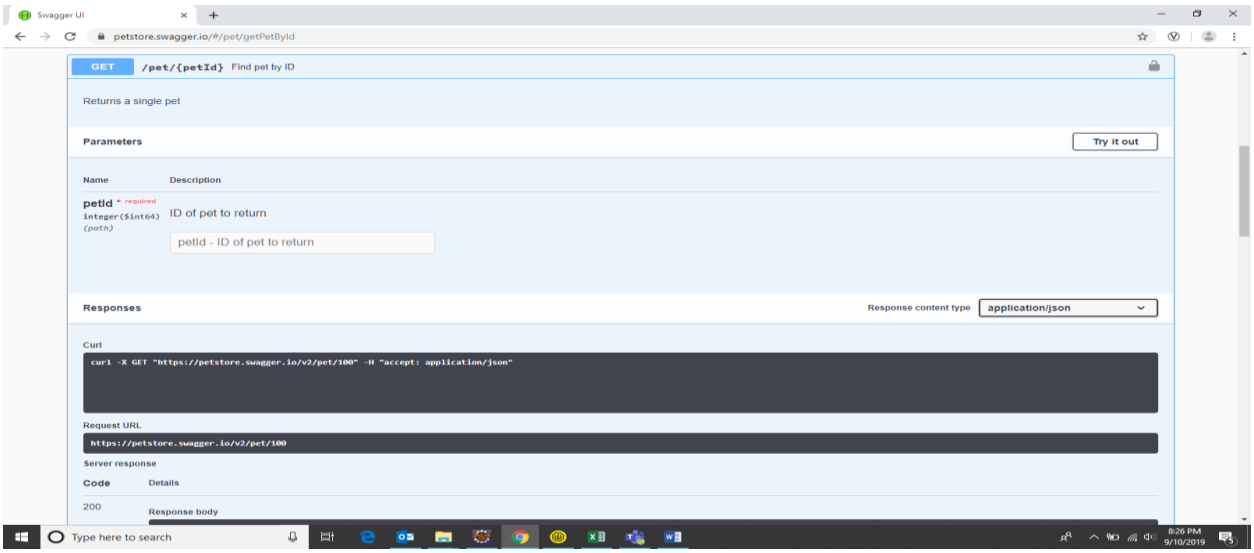
STEP 1:

First go to the given swagger link. You will be redirected to the page as shown in the below figure.



STEP2:

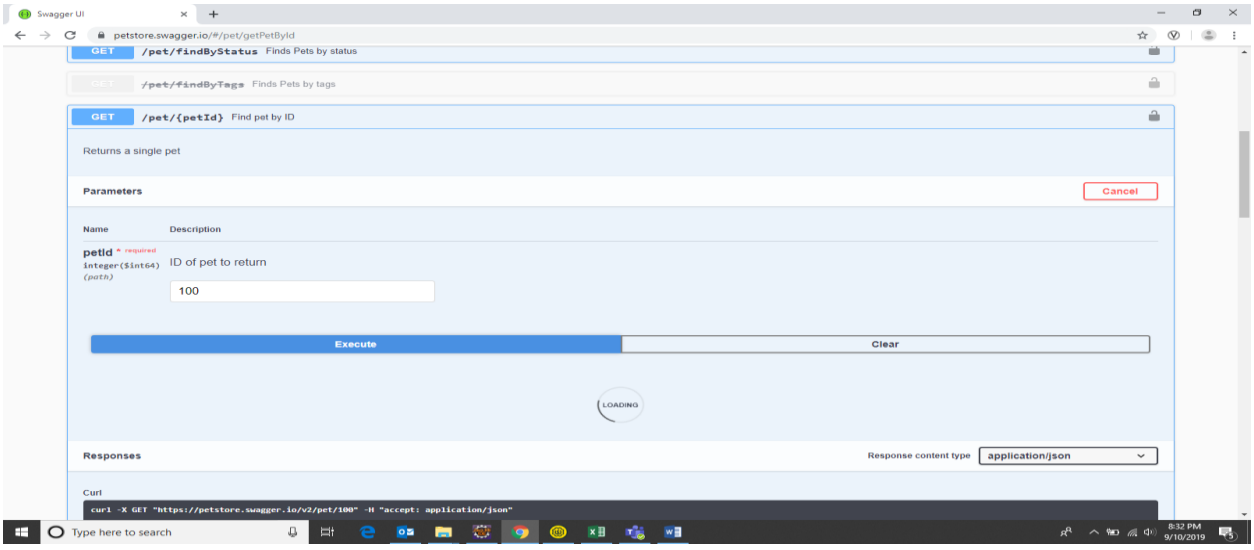
Click on ‘Find Pet by Id’.



STEP 3:

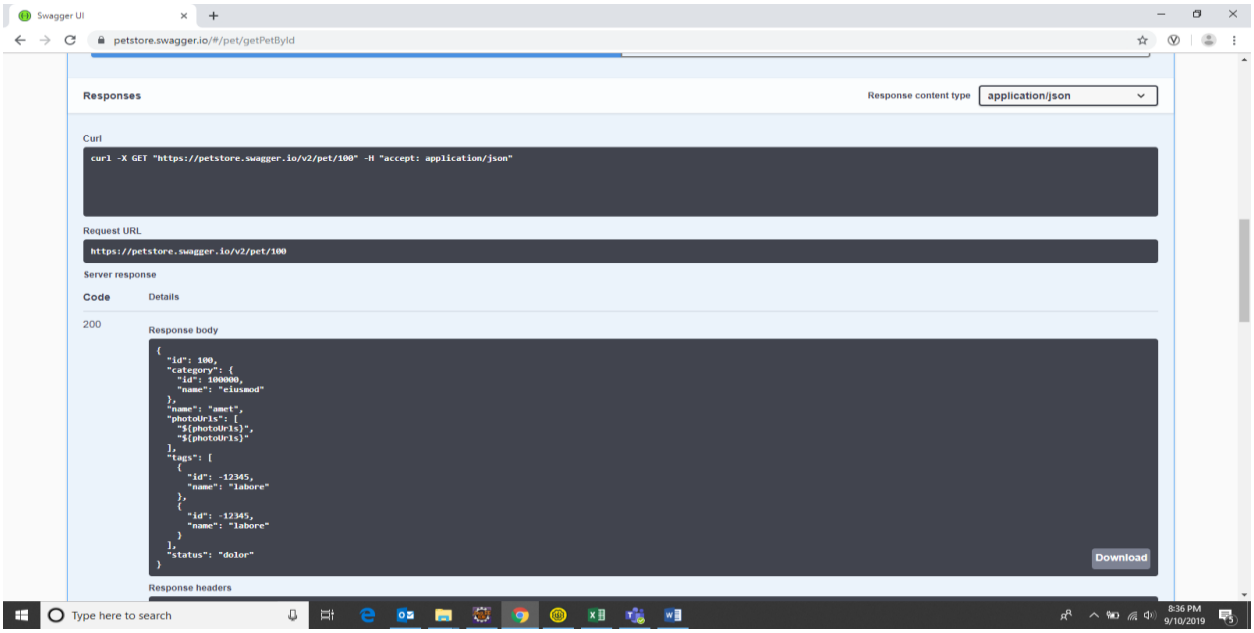
Click on 'Try It Out' and give some pet id to see whether that petid is present or not. If the petId is present then it will give response body with pet details. If that pet id is not present then it will give some error message in the response body.

Change the application/xml to application/json before executing to get the JSON response.



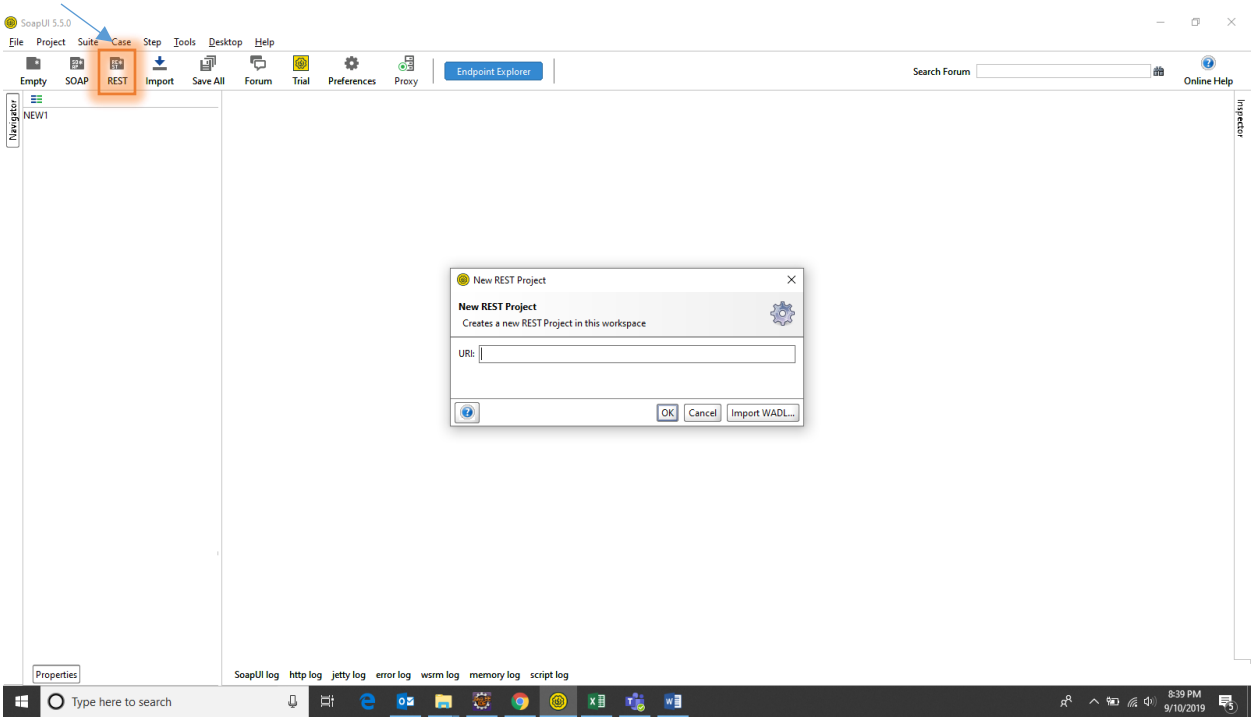
STEP 4:

If the id is present then you will get some JSON response like this:



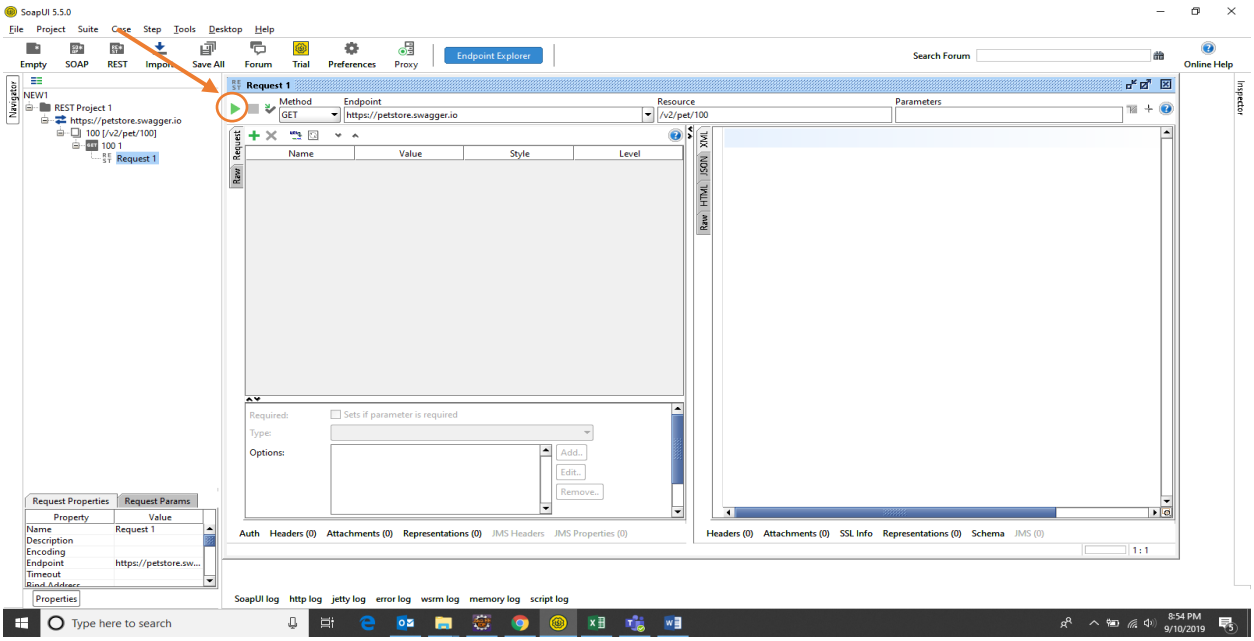
STEP 5:

Copy the request URL and open the SOAP UI application and click on the REST.



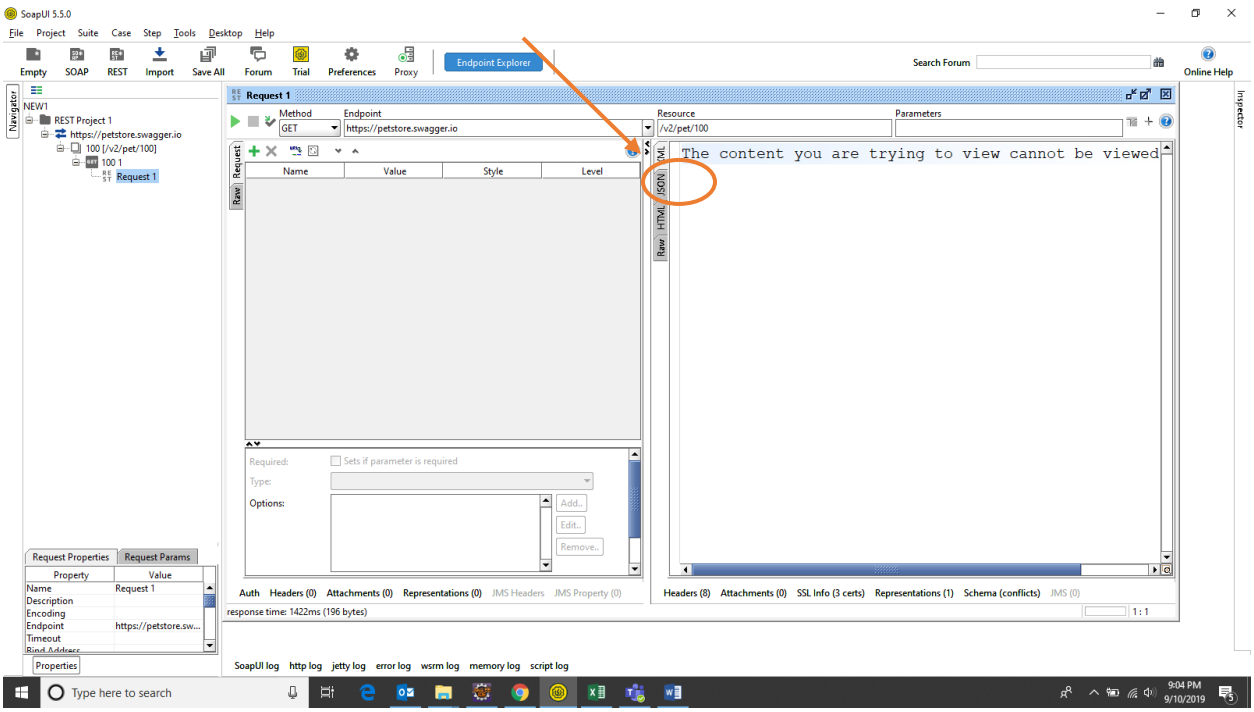
STEP 6:

Paste that GET URL in the URI text box. Then click OK. You will get something as shown in the picture. Then click on the green button to run.



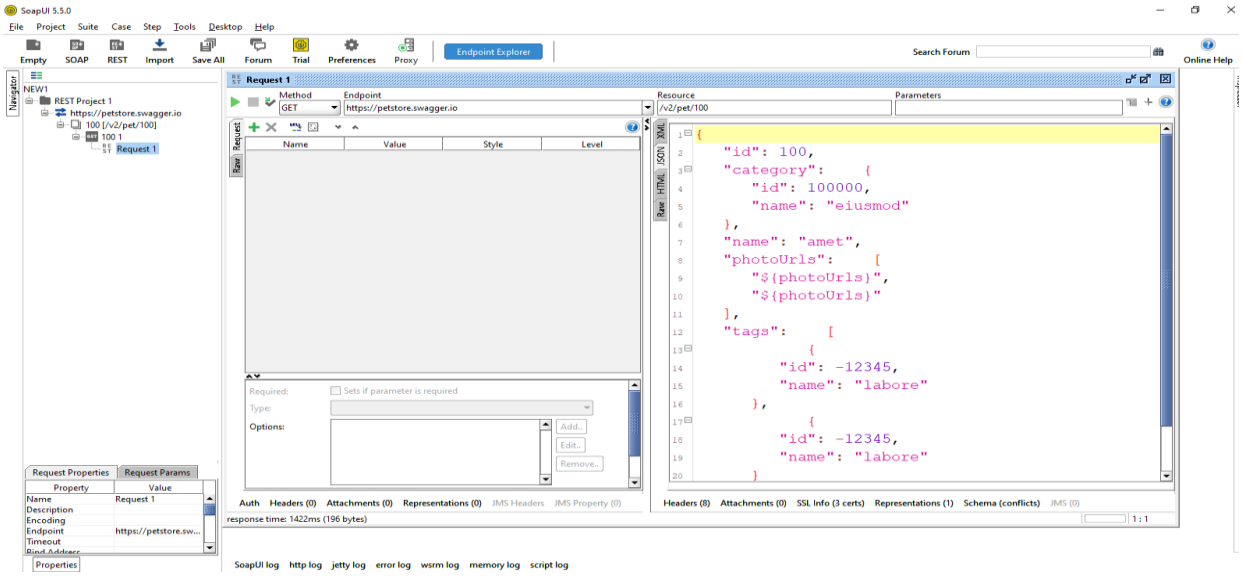
STEP 7:

You will get the response in XML. To get the response in JSON click on the JSON.



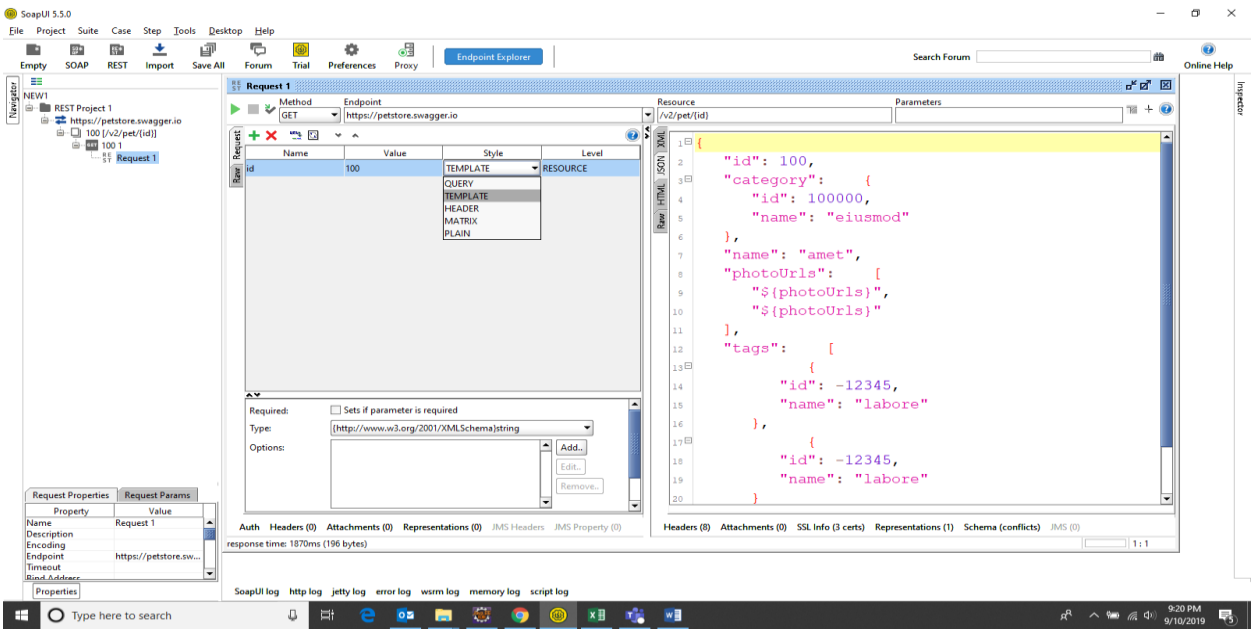
STEP 7:

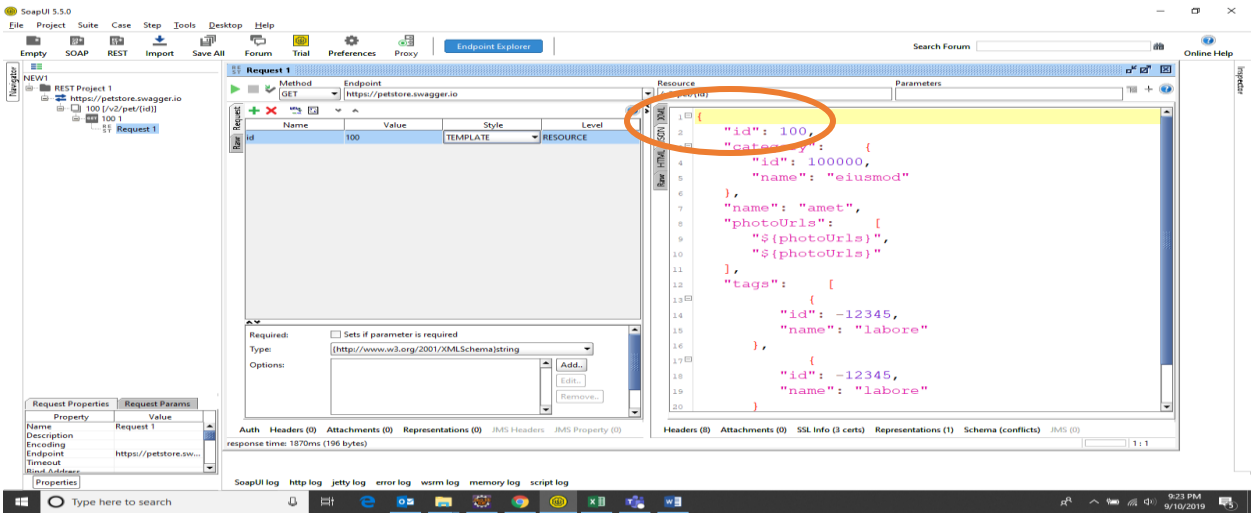
You will get JSON response for GET petid = 100.



STEP 8:

We can also pass the pet id as a parameter. Change the style from query to TEMPLATE as shown in the below figure.





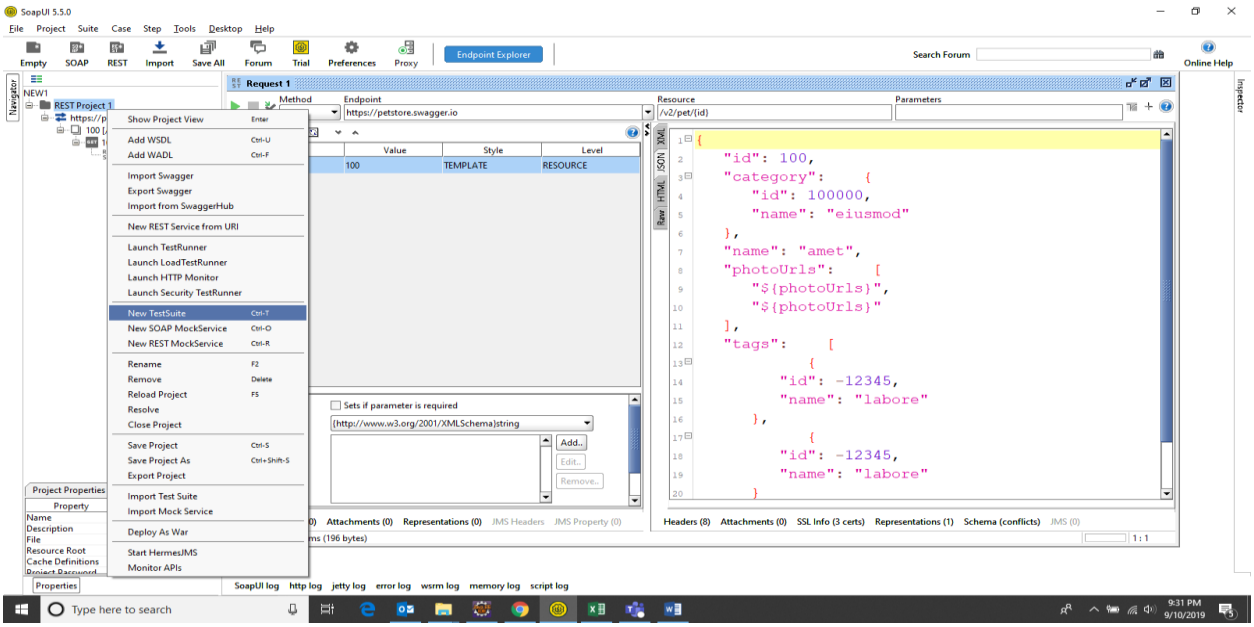
There will be some change in the resources. Resources should contain the parameter name in {}. For the above example it is {id}.

CREATING GROOVY SCRIPT FOR THE GIVEN

REST REQUEST FOR GET

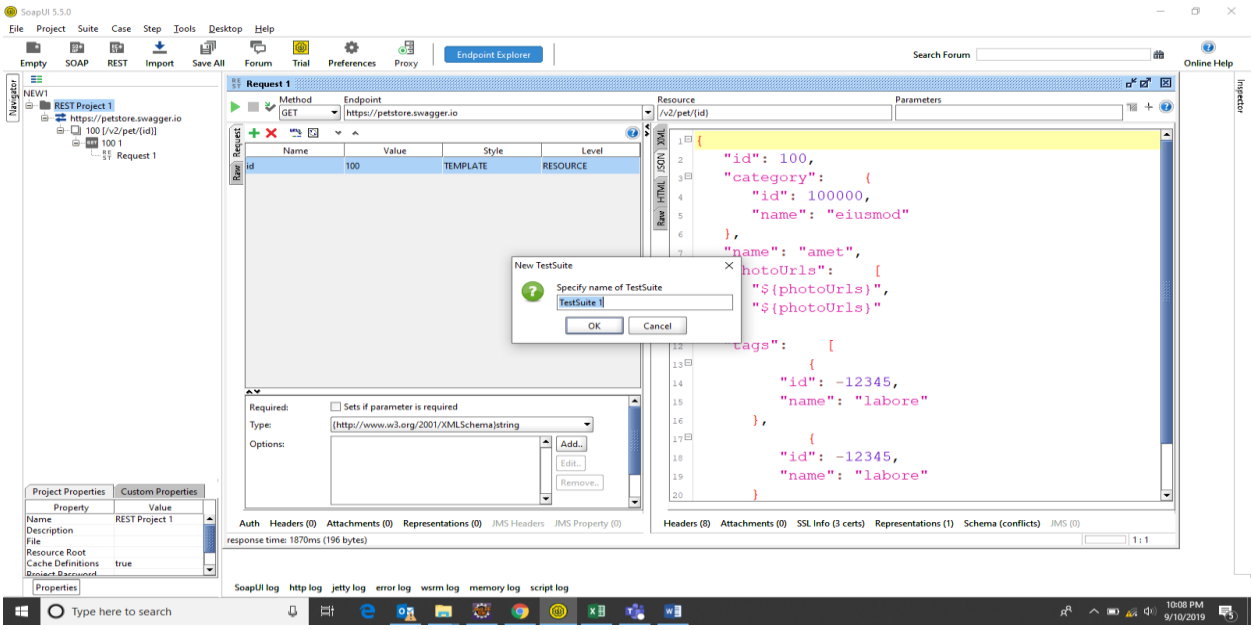
STEP 1:

Right click on the project name (here it is REST Project 1). Then select ‘New TestSuite’.



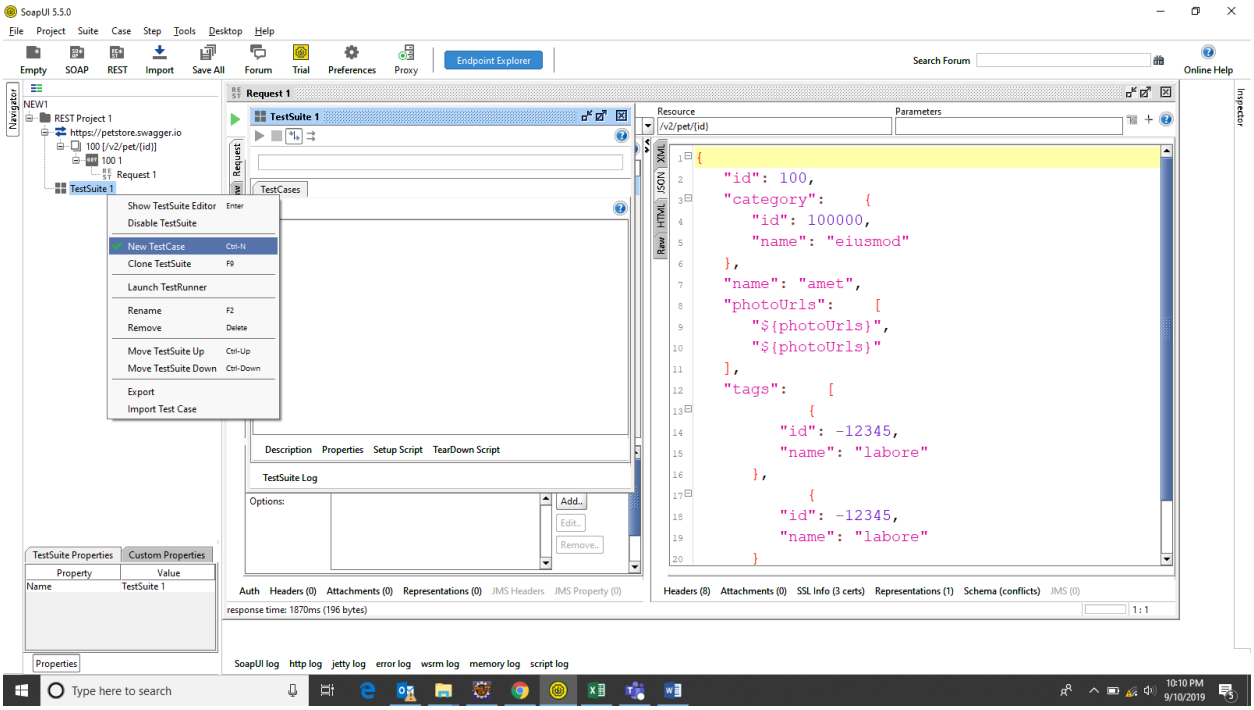
STEP 2:

Give the name for the test suite. Then click ok.



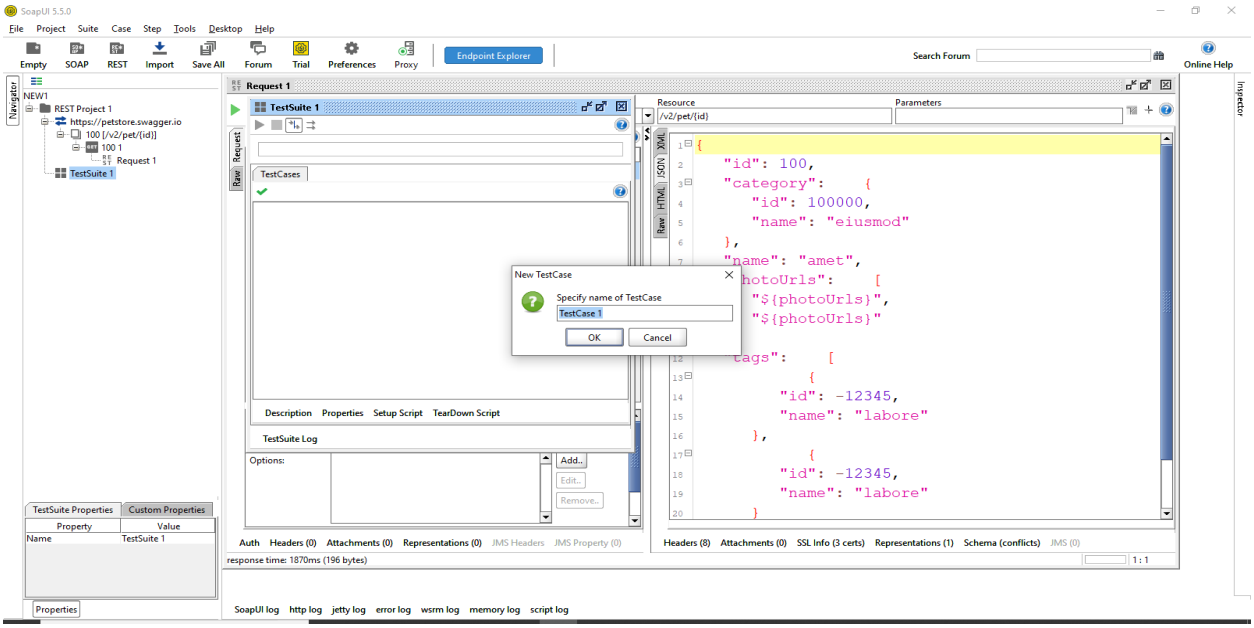
STEP 3:

Right click on the TestSuite and then click New TestCase.



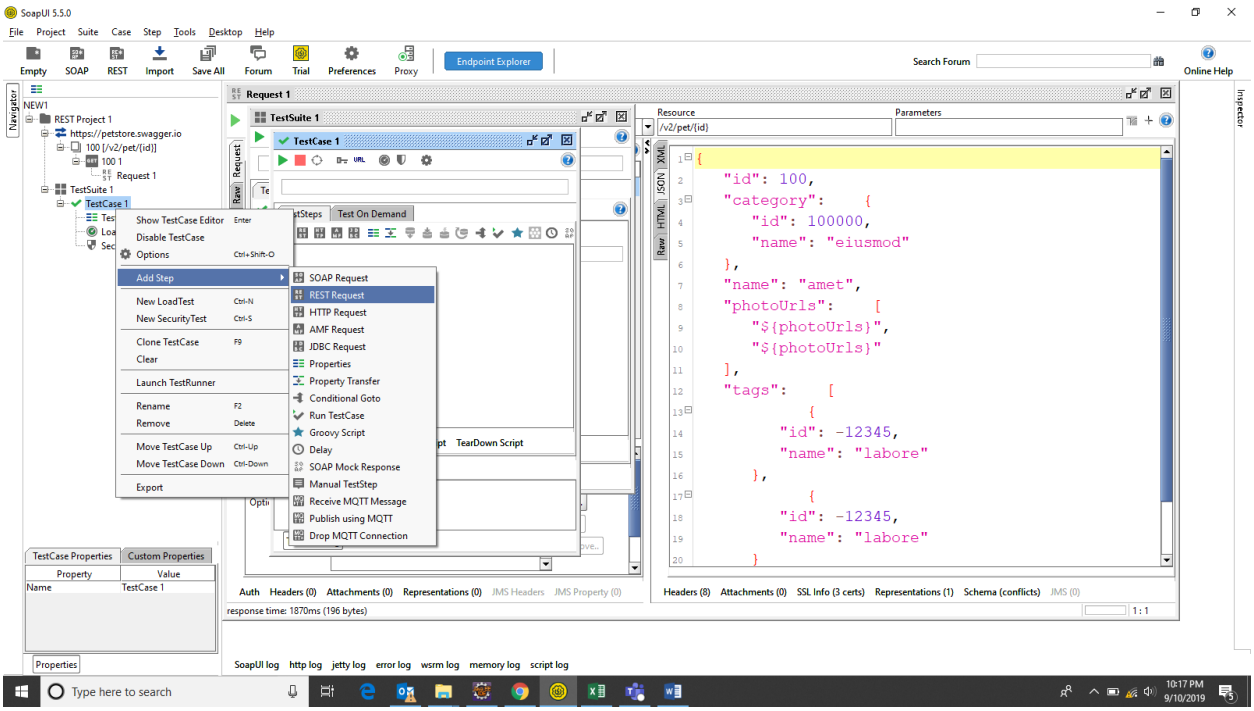
STEP 4:

Give the Test case name and click ok.



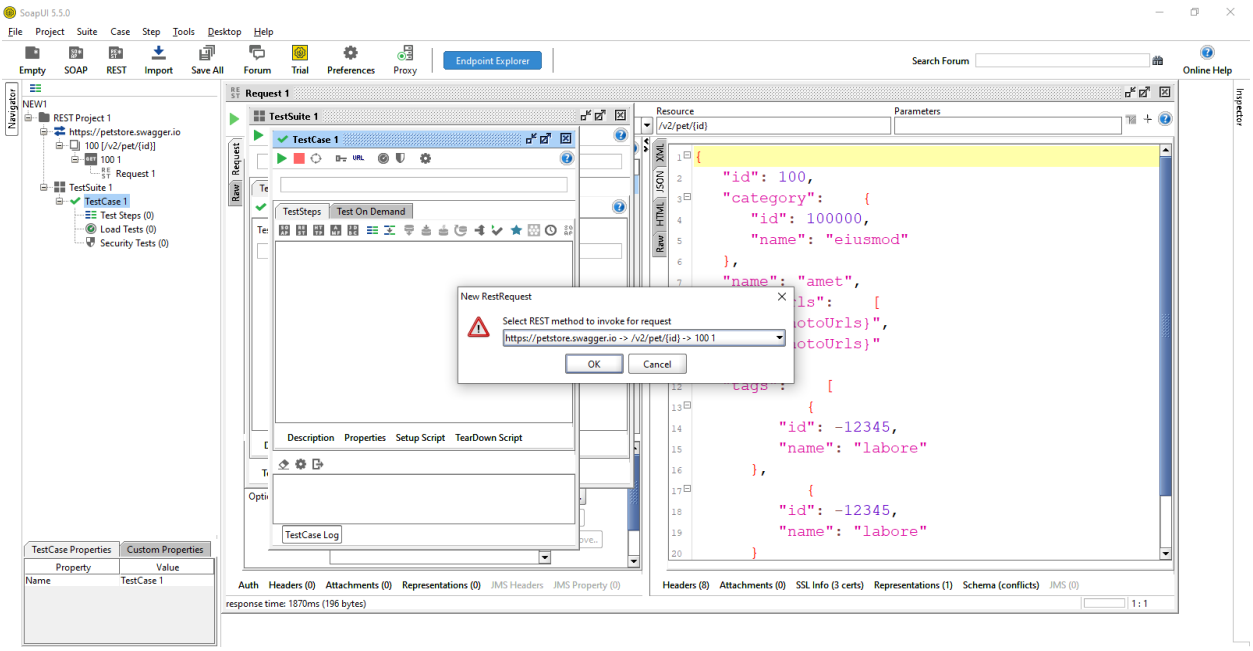
STEP 5:

Right click on the TestCase and select Add Step then select REST Request.



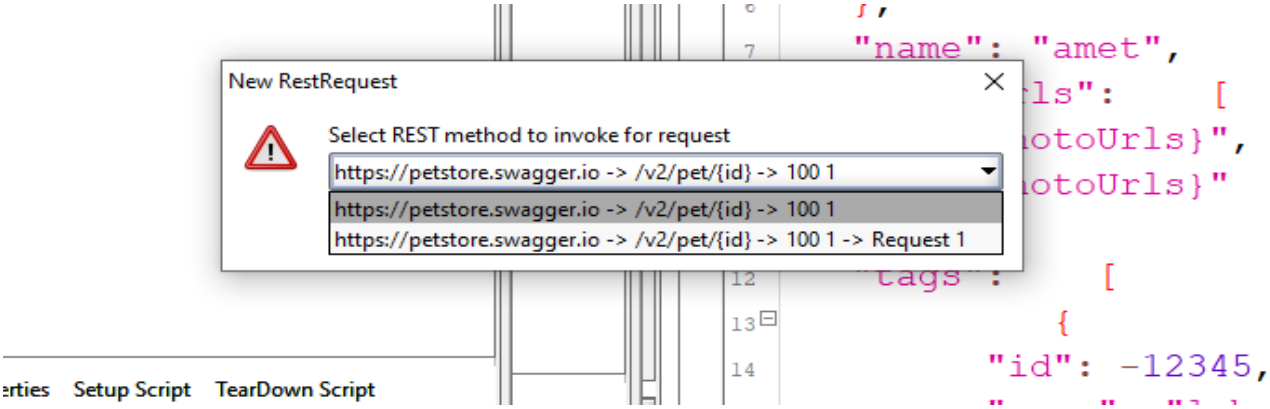
STEP 6:

After giving the name for the REST Request you will get something as shown below:-



STEP 8:

From the drop down list select the link with reference to the GET request which you have created before.

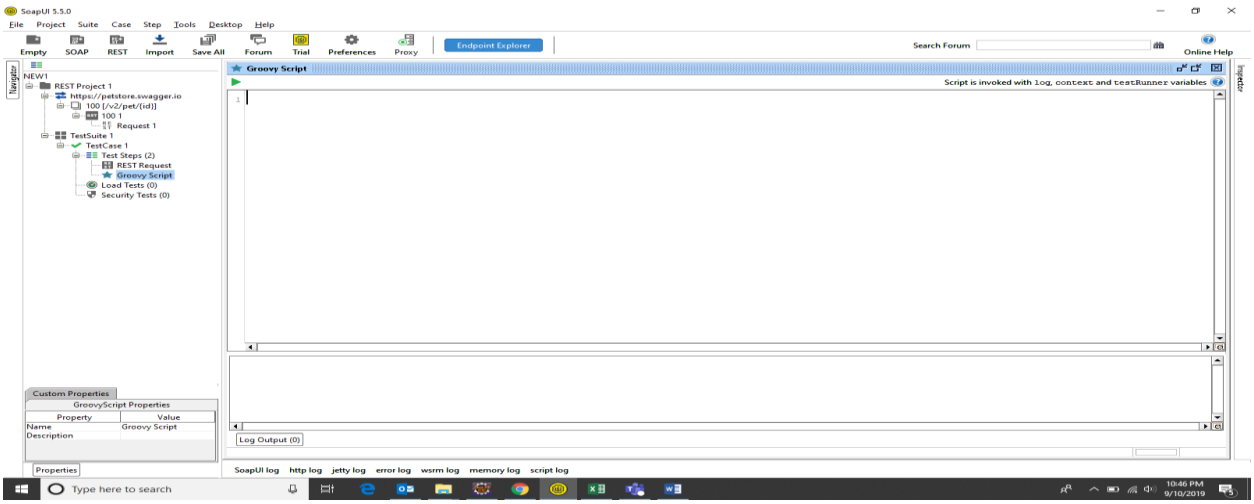


Choose either of the two link and then click ok .

STEP 9:

Now again right click on the test step->add step->Groovy script.

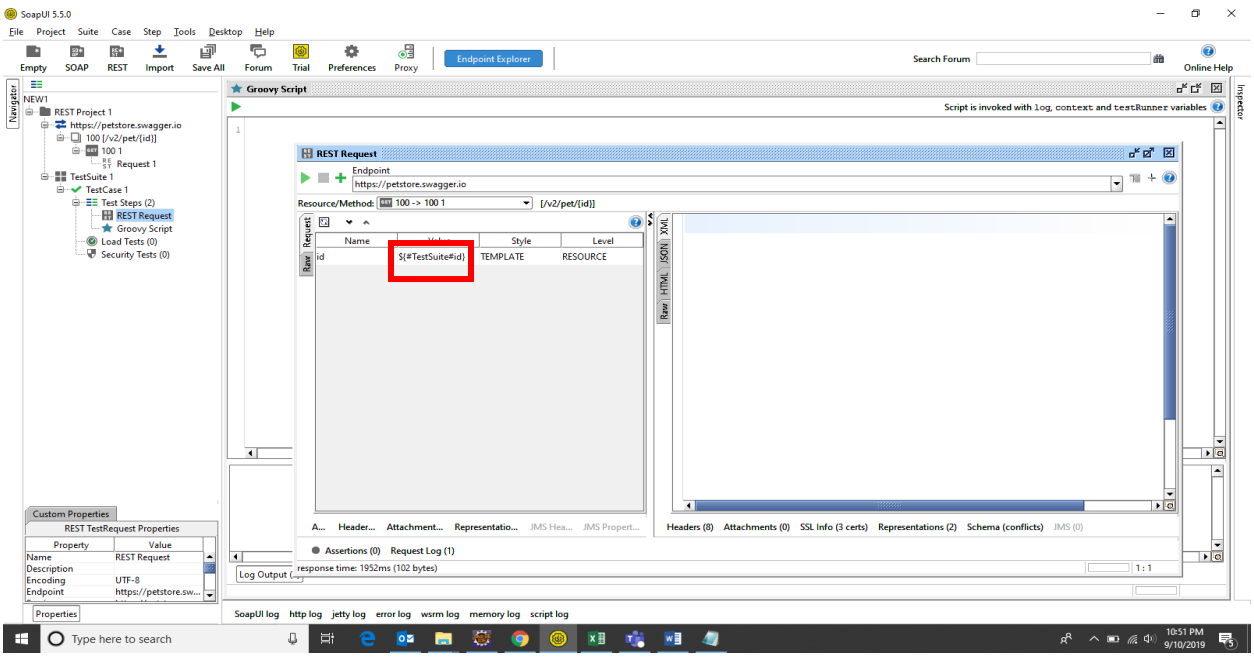
Give the groovy script name and write the code in the groovy script for GET method.



STEP 10:

Give ${#TestSuite#id} in the REST Request at the Test Suite level.

[**note:-**If you want to use the custom property at the Test Case level then give ${#TestCase#attribute_name}]

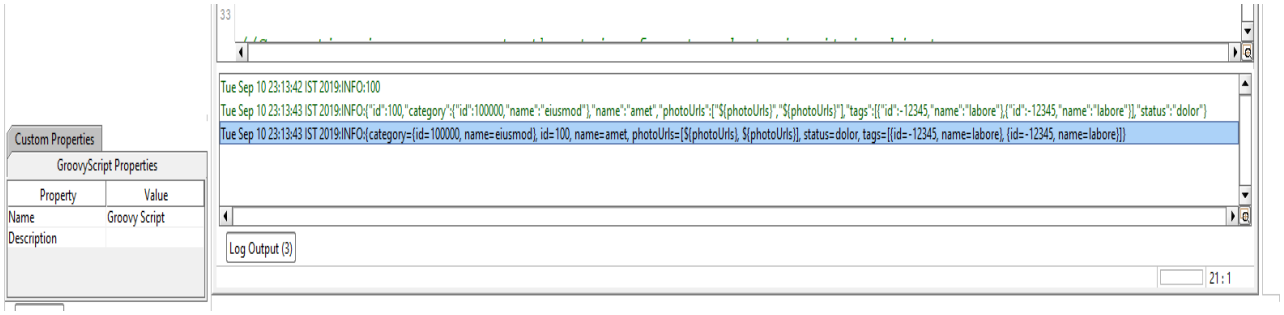


GROOVY SCRIPT CODE FOR GET METHOD

```
import groovy.json.JsonSlurper

//If u want to get the id at TestCase level then dont use .testSuite

// Set a test suite property

testRunner.testCase.testSuite.setPropertyValue("id","100")

// Get a test suite property

def id=testRunner.testCase.testSuite.getPropertyValue("id")

//printing  id

log.info(id)

//Getting the Test Case response. The testcase name should match with the created test case.

//It verifies that which TestCase we want to run.

def tCase = testRunner.testCase.testSuite.testCases["TestCase 1"]

//The REST Request name should match with the created REST Request for get.

//It verifies that which REST Request we want to run.

def getIdTestStep = tCase.testSteps["REST Request"]

//Running the REST request

getIdTestStep.run(testRunner,context)
```

```
//Getting the json response
def responseJson = getIdTestStep.testRequest.response.responseContent

//Printing json response
log.info(responseJson)

//Creating object of JsonSlurper  class
def jsonSlurper = new JsonSlurper()

//Converting json response to the string format and storing it in object
def object = jsonSlurper.parseText(responseJson)

//printing the string response
log.info(object)
```

Output For the Given Example



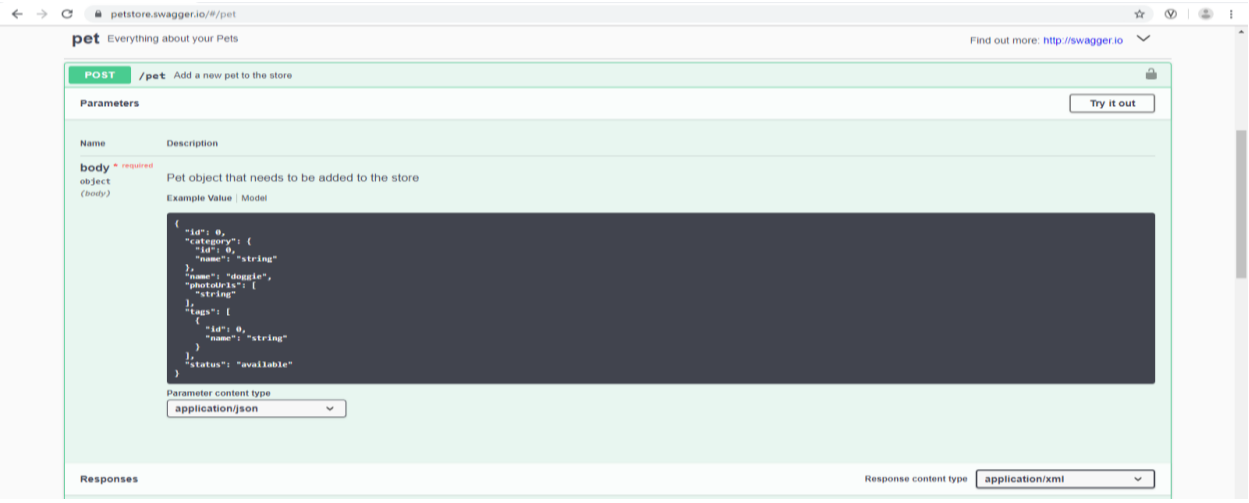
(POST METHOD)

STEP 1:

Again go to the swagger link as provided above.

STEP 2:

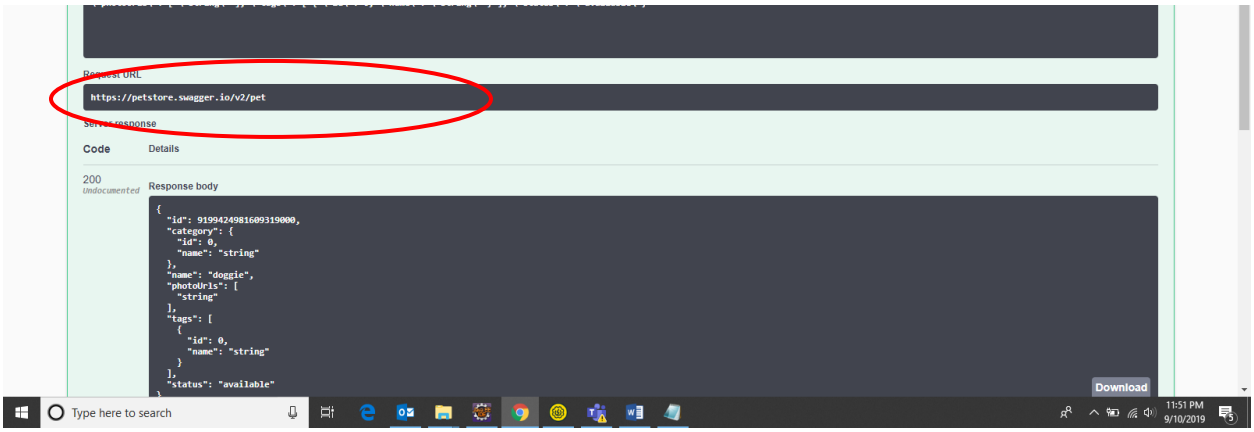
Click on the Post(here,I am using post for “add a new pet to the store”)



STEP 3:

Click on try it out and change to application/json from application/XML.

Execute to see the response body ang to get the End Point for the POST method.

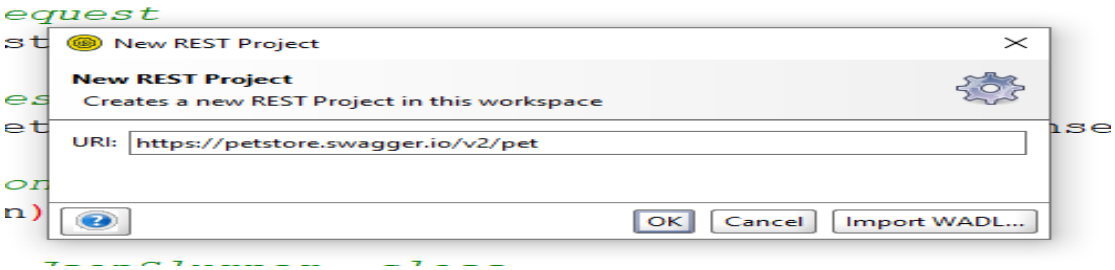


STEP 4:

Now copy the URL for the post.

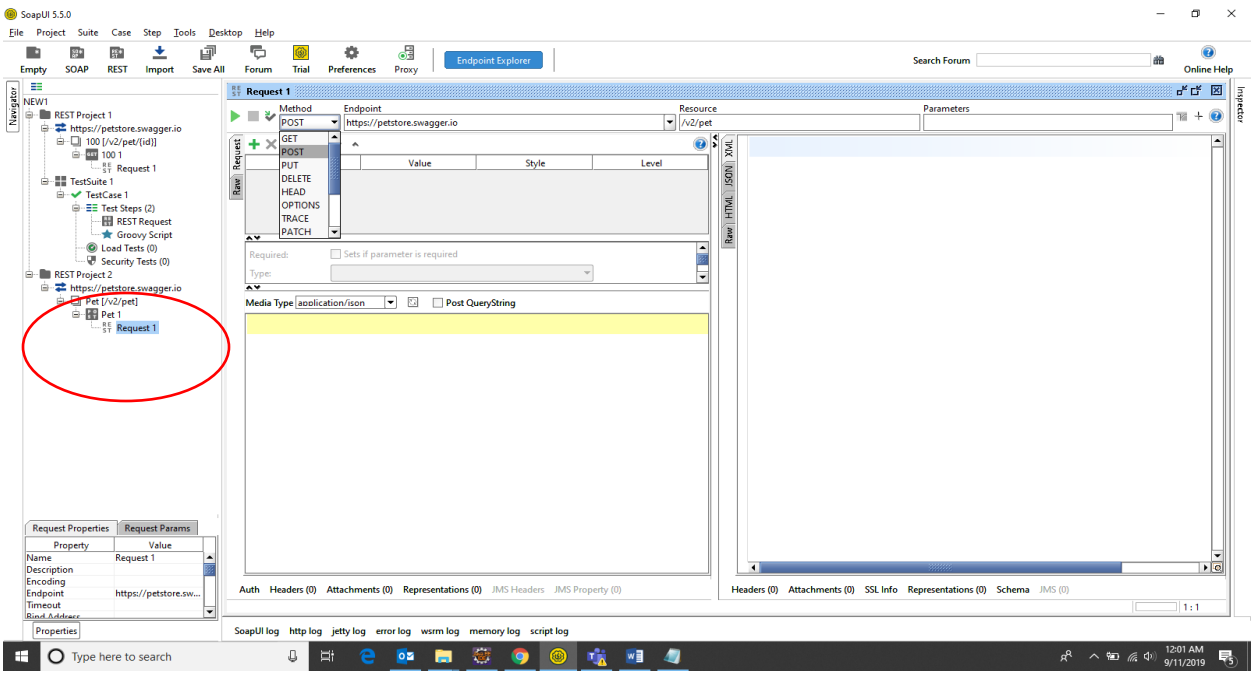
STEP 5:

Create new REST project as created for the GET method and paste the url for the post in the textbox and click ok.(Refer to step 5 of the GET method).



STEP 6:

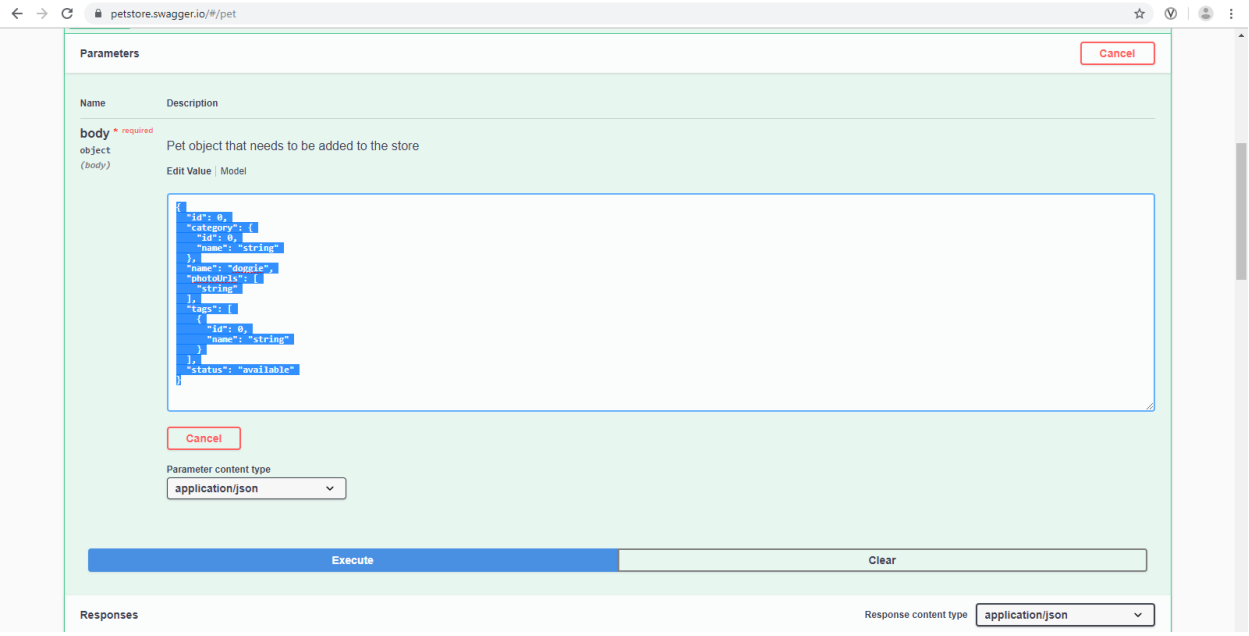
Change the method from GET to POST. Then at the below you will get Media Type.



The marked project is the project created for POST method.

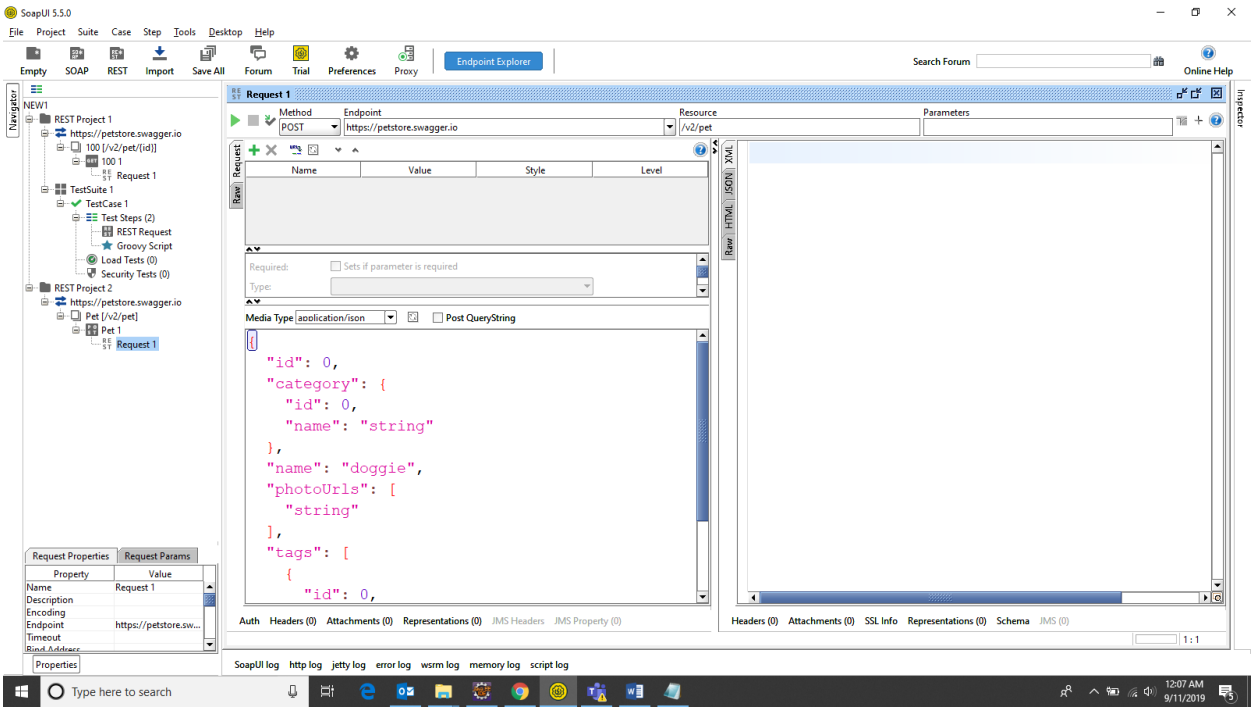
STEP 7:

Now go to the wager link and copy the JSON body for POST.



STEP 8:

Paste this body to the Media Type in your post method.



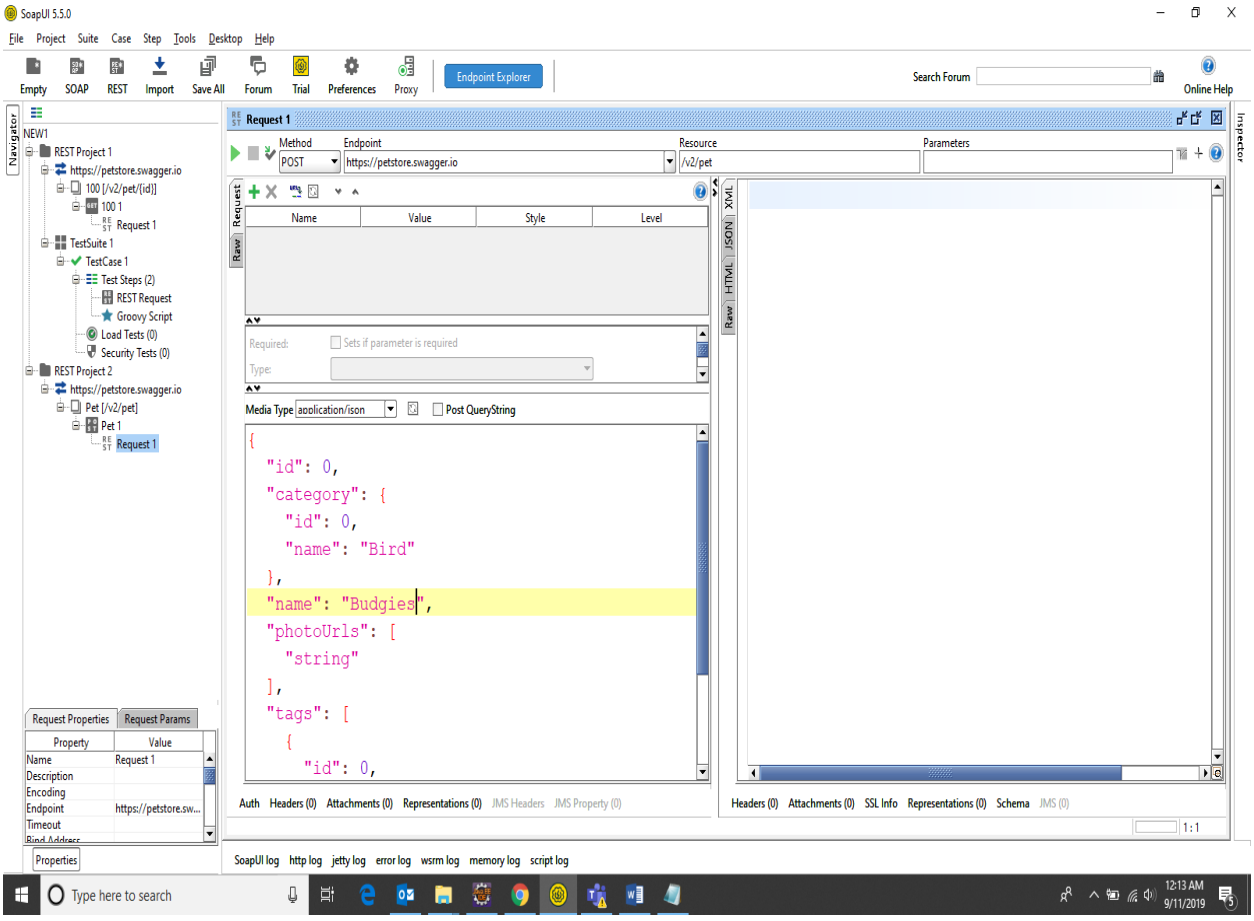
STEP 9:

Now make the necessary changes as required. (Here , for this API Id is auto generated .If we don't make any change then also it will be posted with some new auto generated ID.

But again , if we want to change anything we can change except from the id as It is auto generated for this API.)

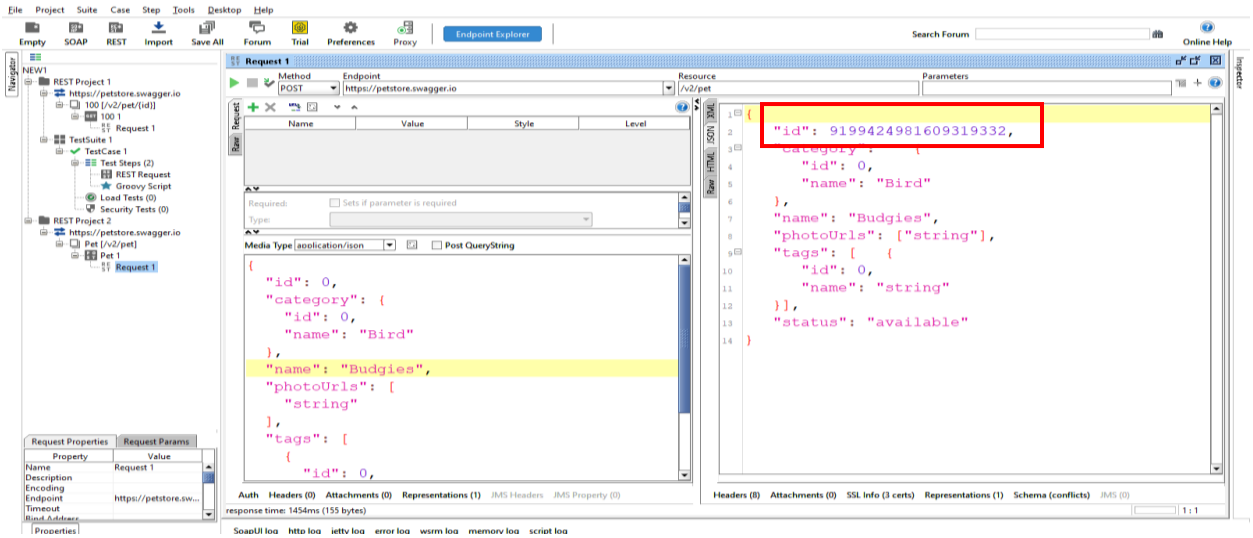
(**note:** For all the API's id is not always auto generated).

Here I am making some of the changes as shown below.



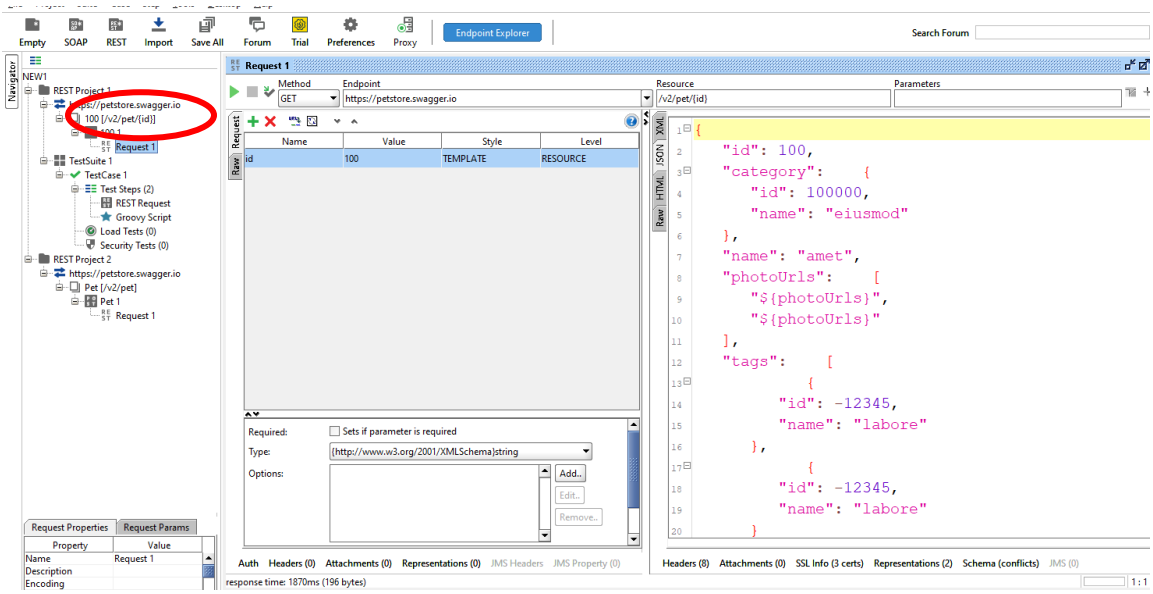
STEP 10:

After making the required changes click on the run button and if it is successfully posted then you will get some auto generated id in the response along with the details.



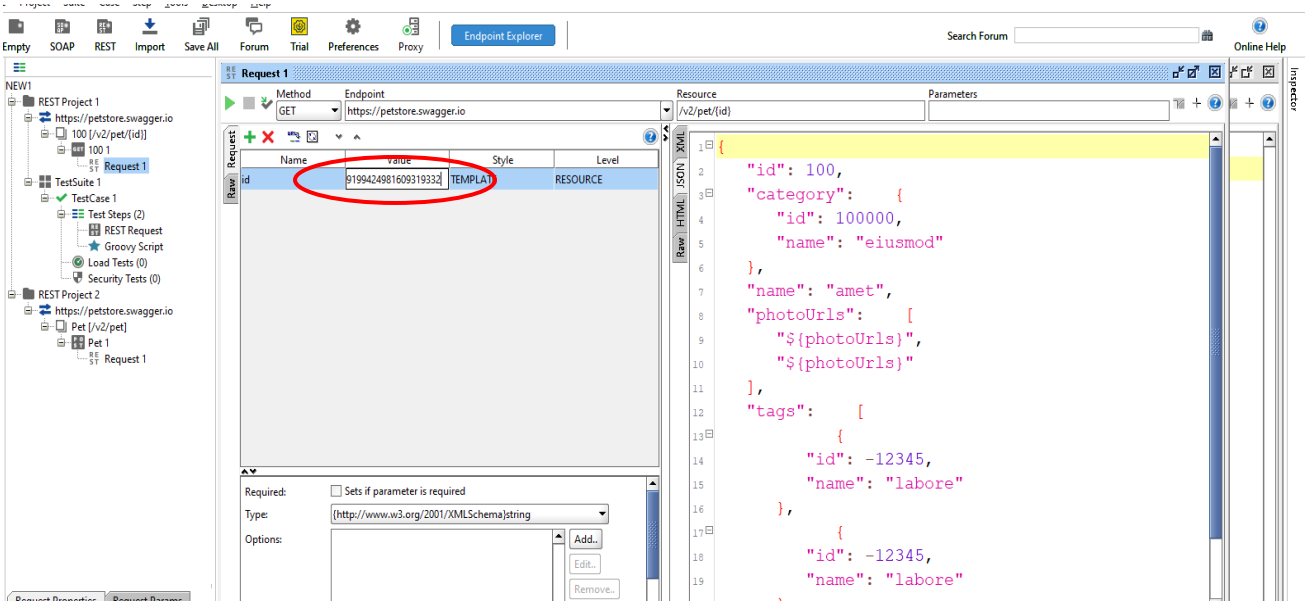
STEP 11:

Again go to the Get Request to see whether that id exist or not.(This step is only to confirm wether it is posted or not).Marked one in the below picture is the mthod for GET.



STEP 11:

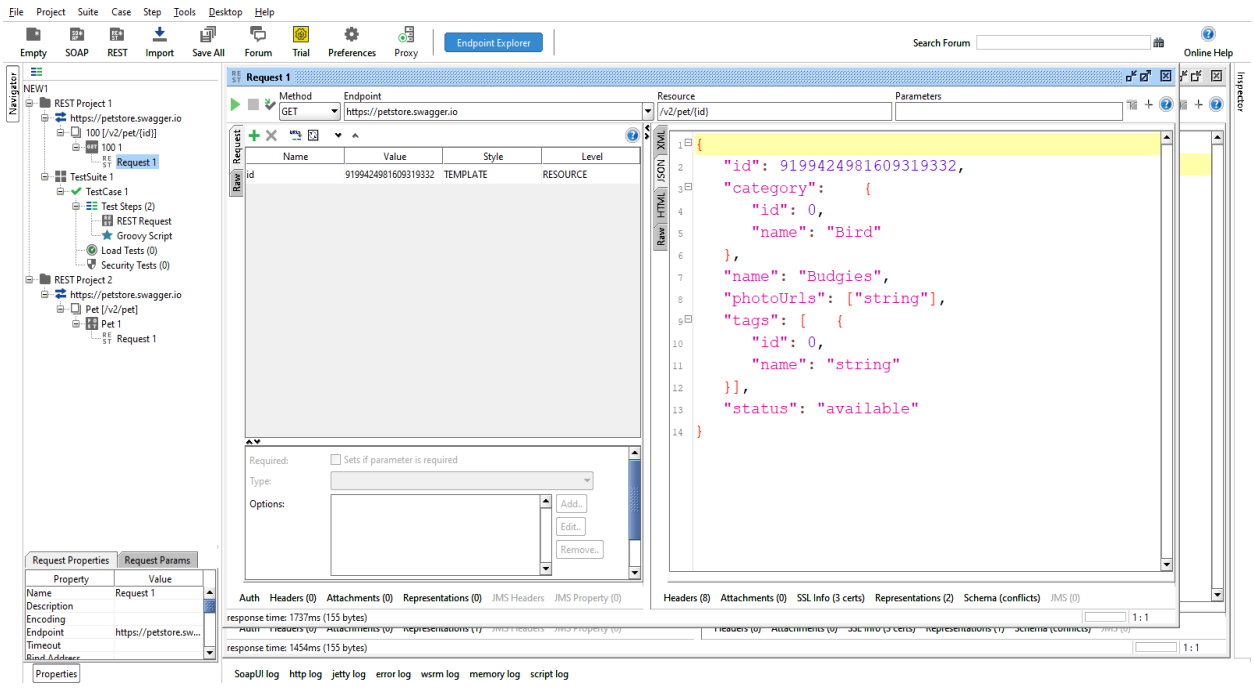
Give the auto generated id in the value.



STEP 12:

Run get request.

You will get the response for that id.



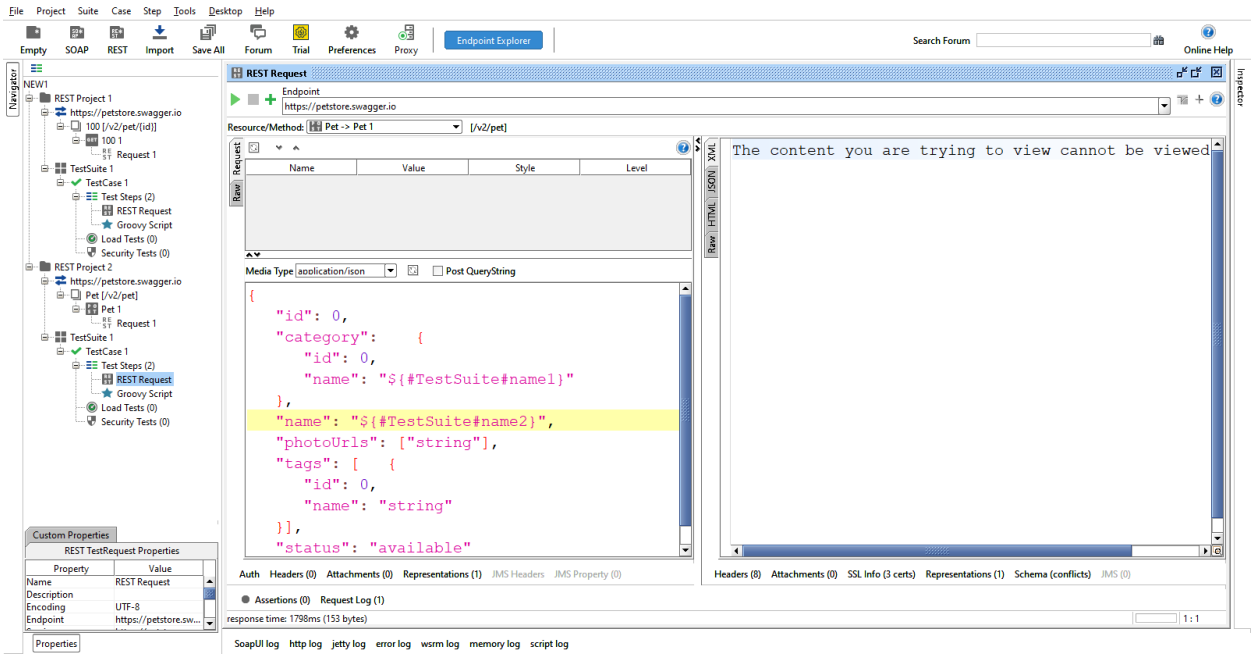
CREATING GROOVY SCRIPT FOR THE GIVEN REST REQUEST FOR GET

STEP 1:

Follow STEP 1,STEP 2,STEP 3,STEP 4,STEP 5,STEP 6,STEP 7,STEP 8,STEP 9 of “ **CREATING GROOVY SCRIPT FOR THE GIVEN REST REQUEST FOR GET”** to create a REST request and groovy script for POST.

STEP 2:

Now copy the JSON body of post from the given swagger link as done in STEP 7 of POST method and paste it in the Media Type of the REST Request.



Any value which you want to pass from the groovy script should be written like

${#TestSuite#attribute_name}.

For this I am passing the value of the first name and the value of the second name from the groovy script.

Groovy Script Code for post Method

```
import groovy.json.JsonSlurper

testRunner.testCase.testSuite.setPropertyValue("name1","pet")

testRunner.testCase.testSuite.setPropertyValue("name2","turtle")

def nametest=testRunner.testCase.testSuite.getPropertyValue("name 1")

def jobtest=testRunner.testCase.testSuite.getPropertyValue("name 2")

def tCase=testRunner.testCase.testSuite.testCases["TestCase 1"]

def addDetails=tCase.testSteps["REST Request"]

addDetails.run(testRunner, context)

def responseJson = addDetails.testRequest.response.responseContent

log.info(responseJson)

def jsonSlurper = new JsonSlurper()

def object = jsonSlurper.parseText(responseJson)
```

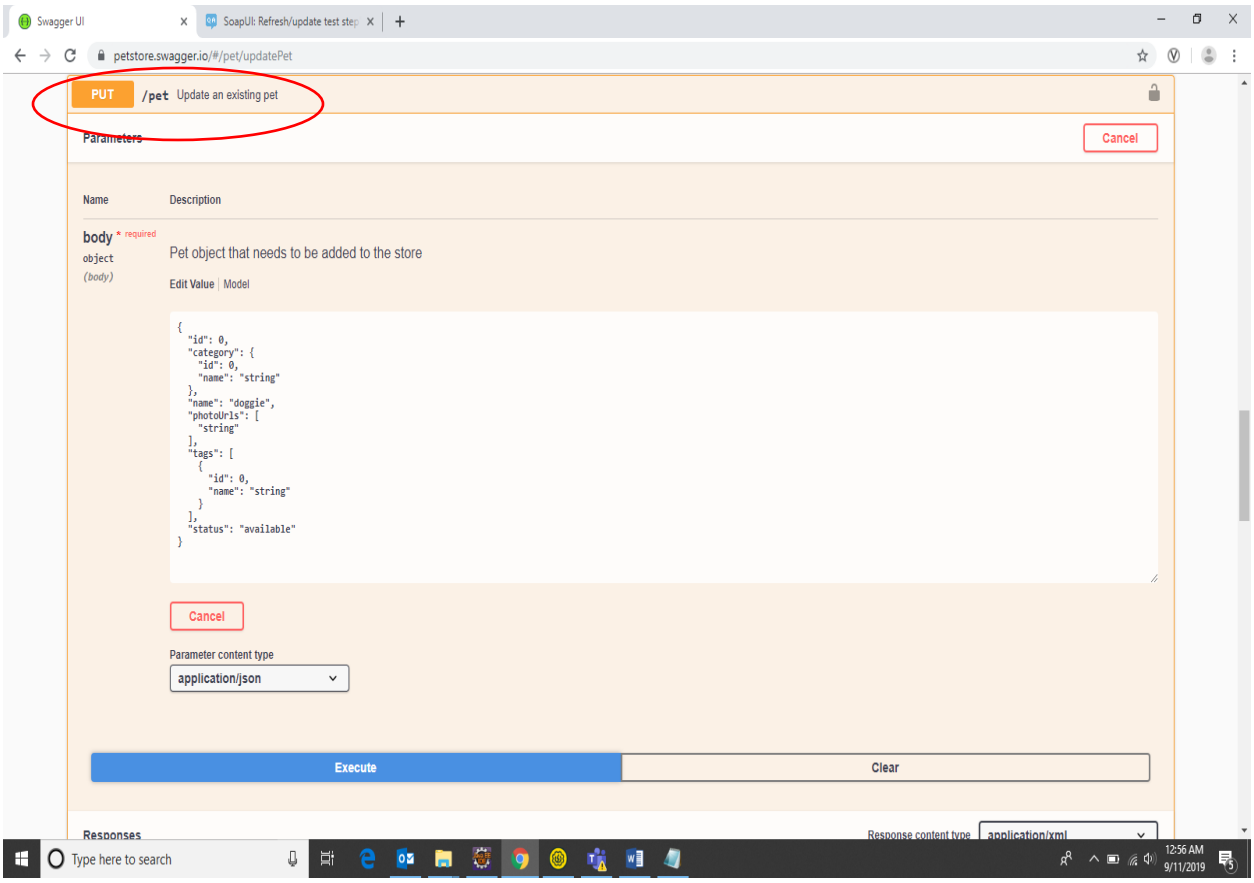
//object.id will print only the id from that response

def id=""+object.id

log.info(id)

(Put Method)

In put method every thing is same as the post method. Just use the URL for the put.



Just give the Id where you want to update.

CODE FOR PUT METHOD IN GROOVY

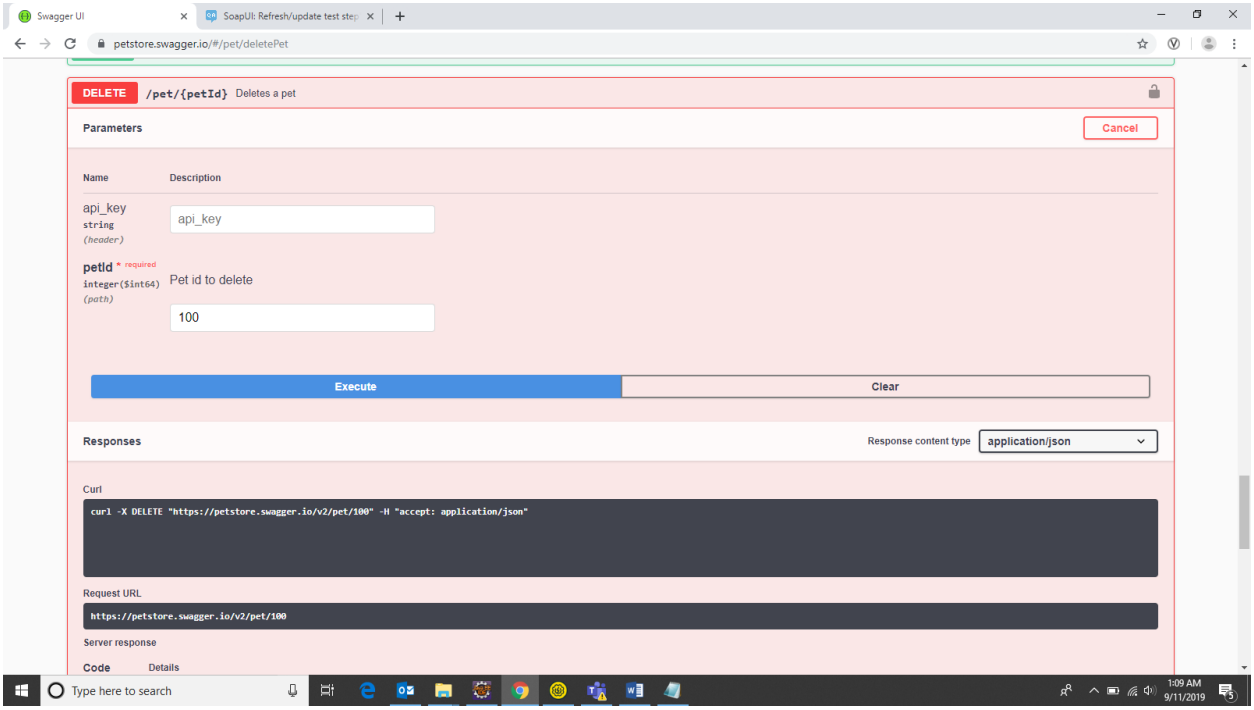
```
import groovy.json.JsonSlurper

testRunner.testCase.testSuite.setPropertyValue("id"," 91994249816093193 56")

testRunner.testCase.testSuite.setPropertyValue("name1","pet1")

testRunner.testCase.testSuite.setPropertyValue("name2","birds")


def idpet=testRunner.testCase.testSuite.getPropertyValue("id")


def petname1=testRunner.testCase.testSuite.getPropertyValue("name1")


def petname2= testRunner.testCase.testSuite.getPropertyValue("name2")



def tCase=testRunner.testCase.testSuite.testCases["TestCase"]


def addDetails=tCase.testSteps["REST Request"]

addDetails.run(testRunner, context)

def responseJson = addDetails.testRequest.response.responseContent


log.info(responseJson)


def jsonSlurper = new JsonSlurper()


def object = jsonSlurper.parseText(responseJson)
```

def id=""+object.id

log.info(id)

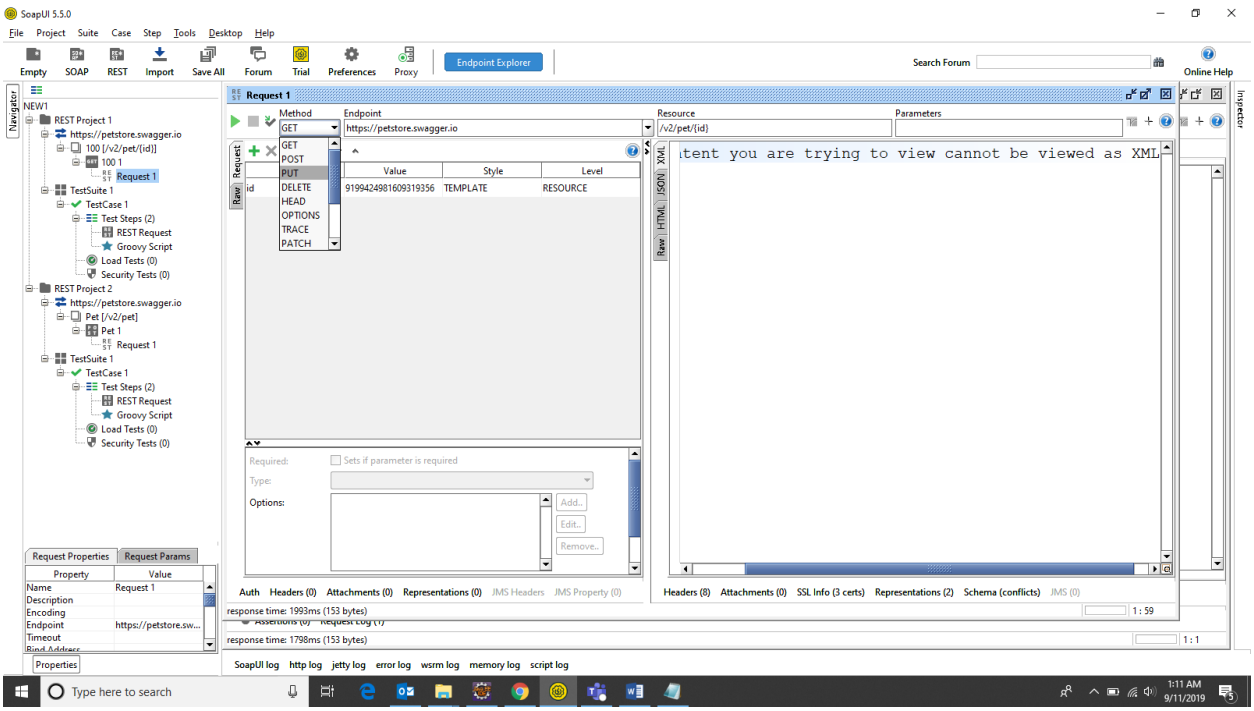
creating REST Request for Delete Method

Just create the new Project and enter the URL for the delete method.



Here the parameter passing is same as that of the GET method.

Do all the steps required for the get method just choose PUT from the dropdown.



Groovy Script Code for post Method

```
testRunner.testCase.testSuite.setPropertyValue("id","100")

def id = testRunner.testCase.testSuite.getPropertyValue("id")

log.info(id)

def tCase = testRunner.testCase.testSuite.testCases["TestCase"]

def delTestStep = tCase.testSteps["REST Request"]
```

```
delTestStep.run(testRunner, context)
```

```
log.info(delTestStep.run(testRunner, context))
```