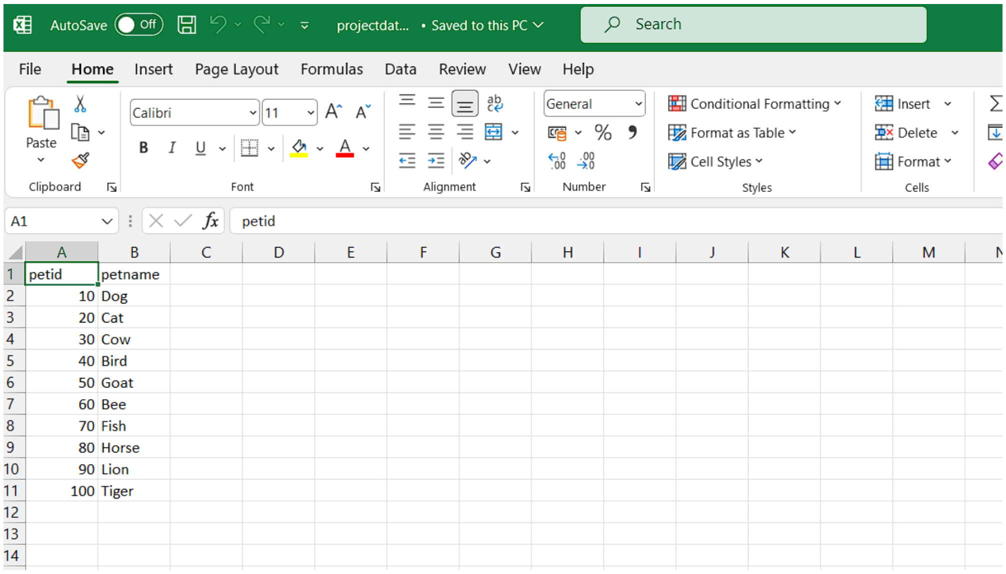
Phase-end Project 3: **To automate functionalities using https://petstore.swagger.io/ REST API services**

(Source code):

Postman Assignment 001:

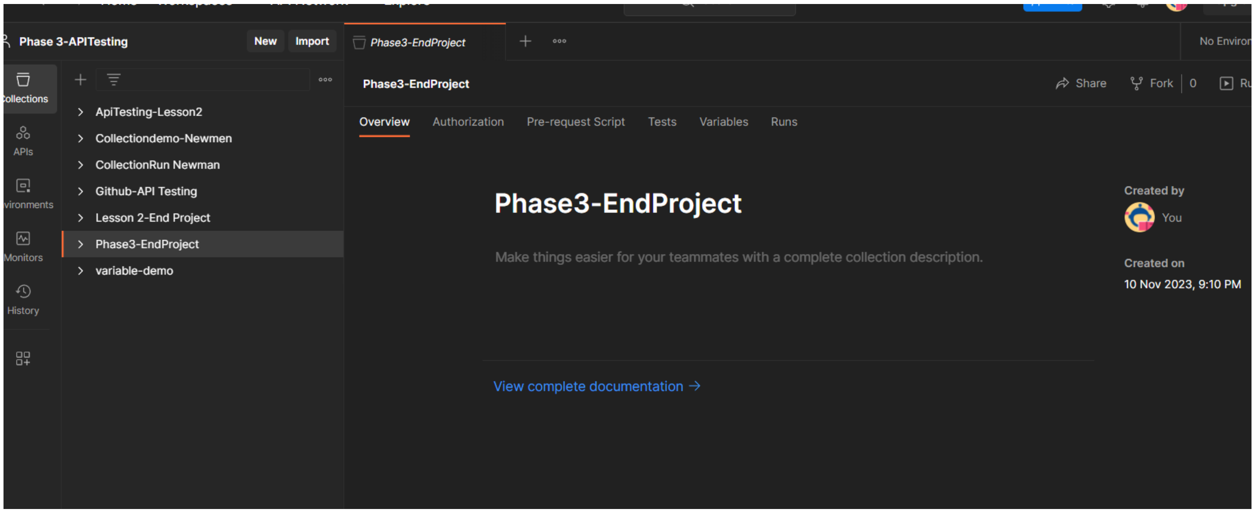
Create a petID and petName:

First, we have to create a excel sheet and enter petID and petName as shown below:



	A	B
1	petid	petname
2	10	Dog
3	20	Cat
4	30	Cow
5	40	Bird
6	50	Goat
7	60	Bee
8	70	Fish
9	80	Horse
10	90	Lion
11	100	Tiger
12		
13		
14		

Then in next step we have to save the above file as .csv and after that open postman and create a collection with name Phase3-EndProject



- ApiTesting-Lesson2
- Collectiondemo-Newmen
- CollectionRun Newman
- Github-API Testing
- Lesson 2-End Project
- Phase3-EndProject
- variable-demo

Phase3-EndProject

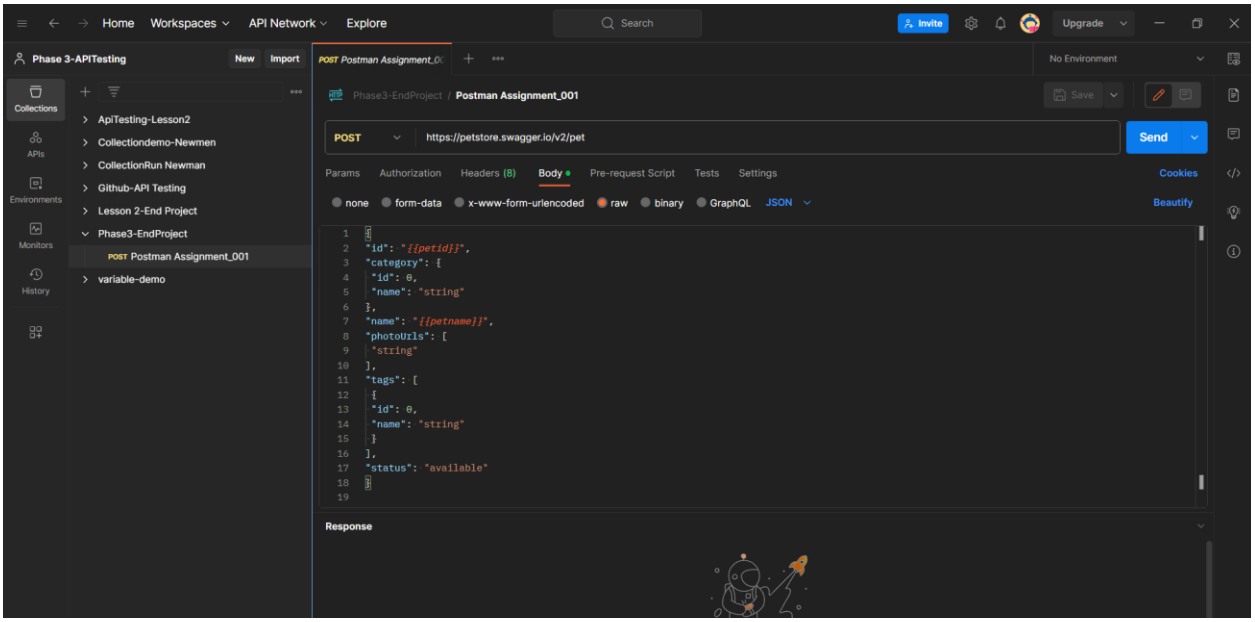
Make things easier for your teammates with a complete collection description.

Created by
You

Created on
10 Nov 2023, 9:10 PM

View complete documentation →

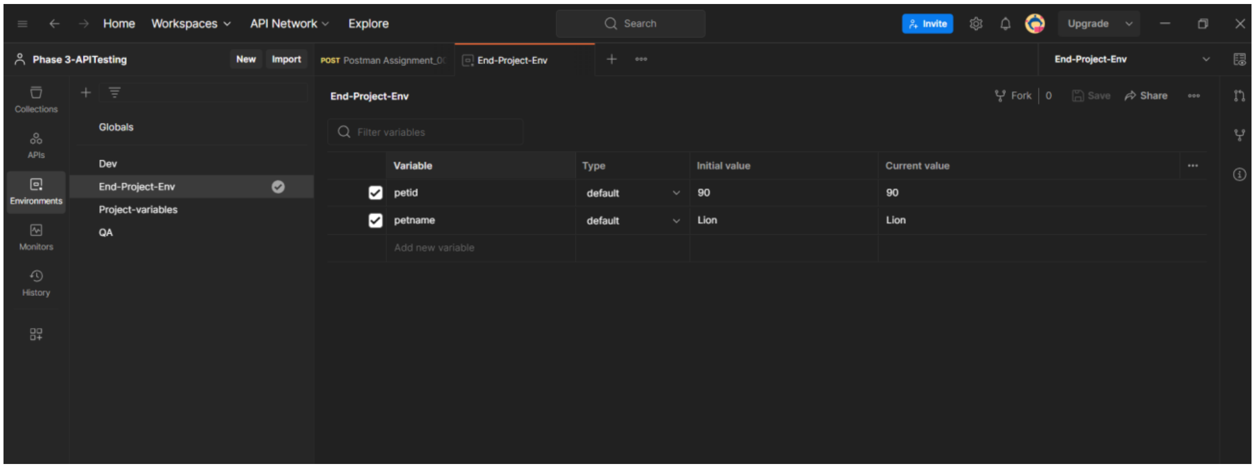
Then we have to Add request as shown below, However according to the Assignment_001-Post we need to use URL as https://petstore.swagger.io/v2/pet and in the body part we have to go to Body -> raw -> JSON and paste the Body as below



POST https://petstore.swagger.io/v2/pet

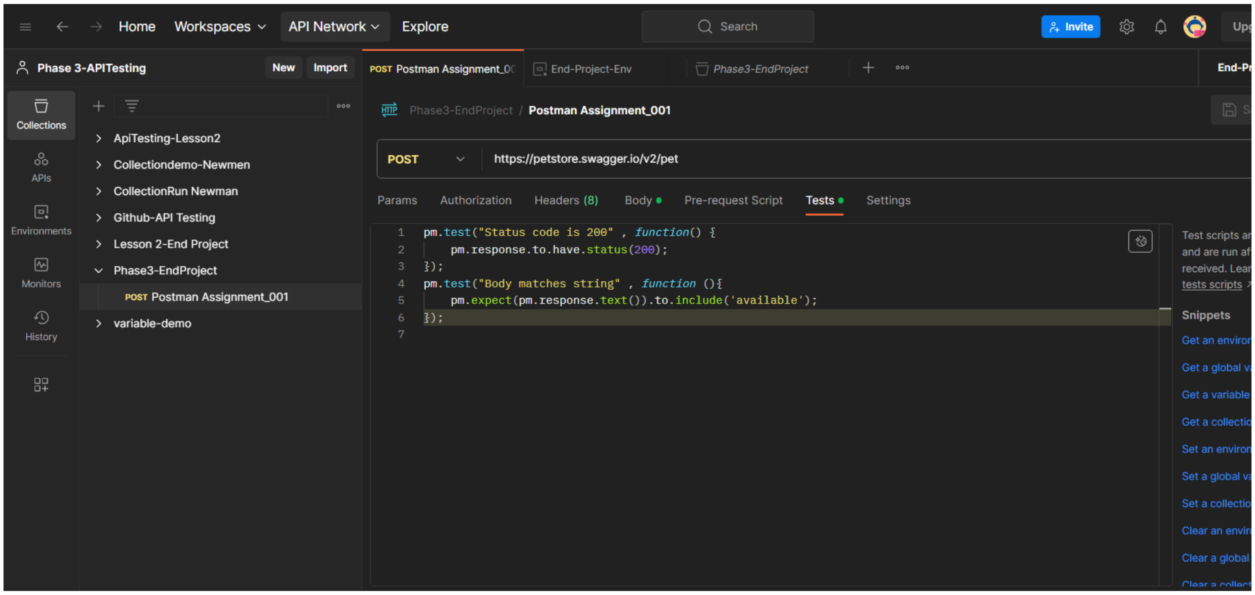
```
{
"id": "{{petid}}",
"category": {
 "id": 0,
 "name": "string"
},
"name": "{{petname}}",
"photoUrls": [
 "string"
],
"tags": [
 {
 "id": 0,
 "name": "string"
 }
],
"status": "available"
}
```

Then create an Environment with name End-Project-Env with two variables petid and petname as shown below



End-Project-Env

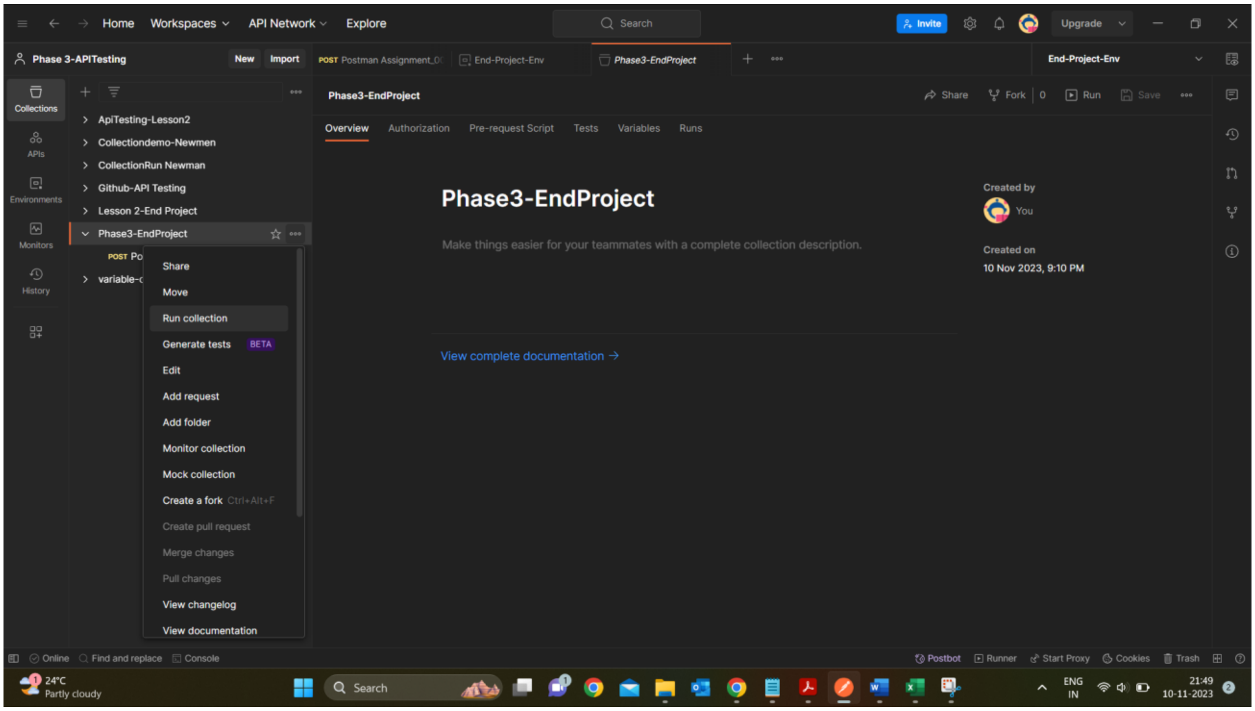
	Variable	Type	Initial value	Current value
☑	petid	default	90	90
☑	petname	default	Lion	Lion



POST https://petstore.swagger.io/v2/pet

```
pm.test("Status code is 200" , function() {
    pm.response.to.have.status(200);
});
pm.test("Body matches string" , function (){
    pm.expect(pm.response.text()).to.include('available');
});
```

Then right click on collection (Phase3-EndProject) and click on Run Collection

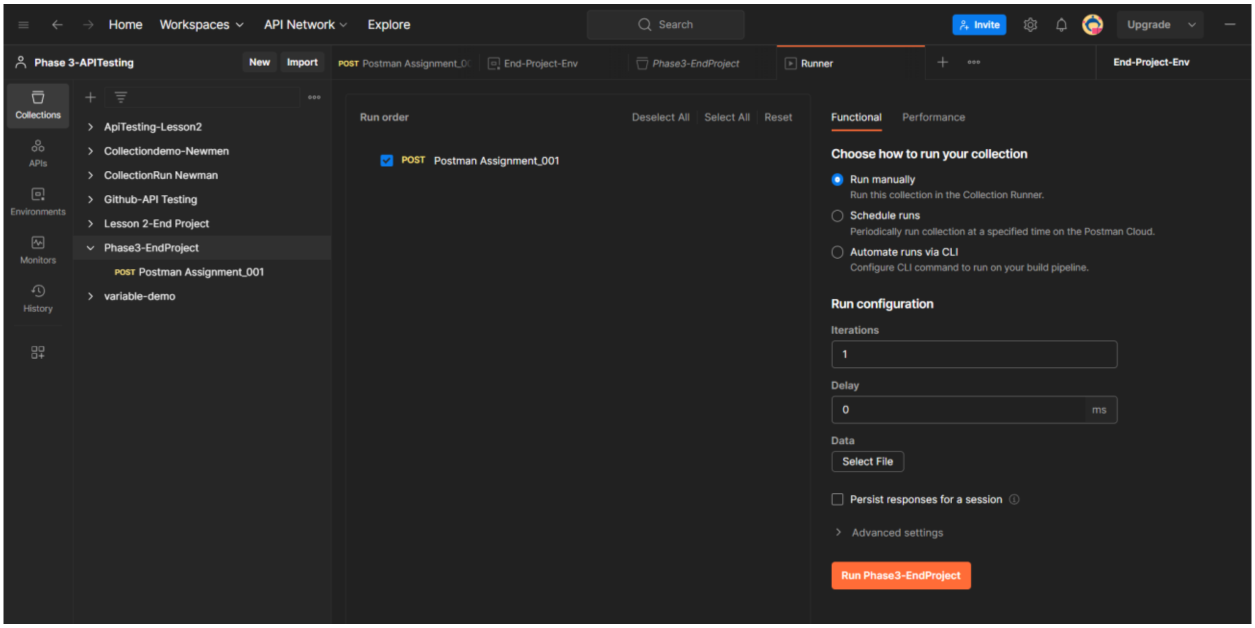


Phase3-EndProject

Make things easier for your teammates with a complete collection description.

Created by
You

Created on
10 Nov 2023, 9:10 PM



Run order

☑ POST Postman Assignment_001

Choose how to run your collection

● Run manually
Run this collection in the Collection Runner.

○ Schedule runs
Periodically run collection at a specified time on the Postman Cloud.

○ Automate runs via CLI
Configure CLI command to run on your build pipeline.

Run configuration

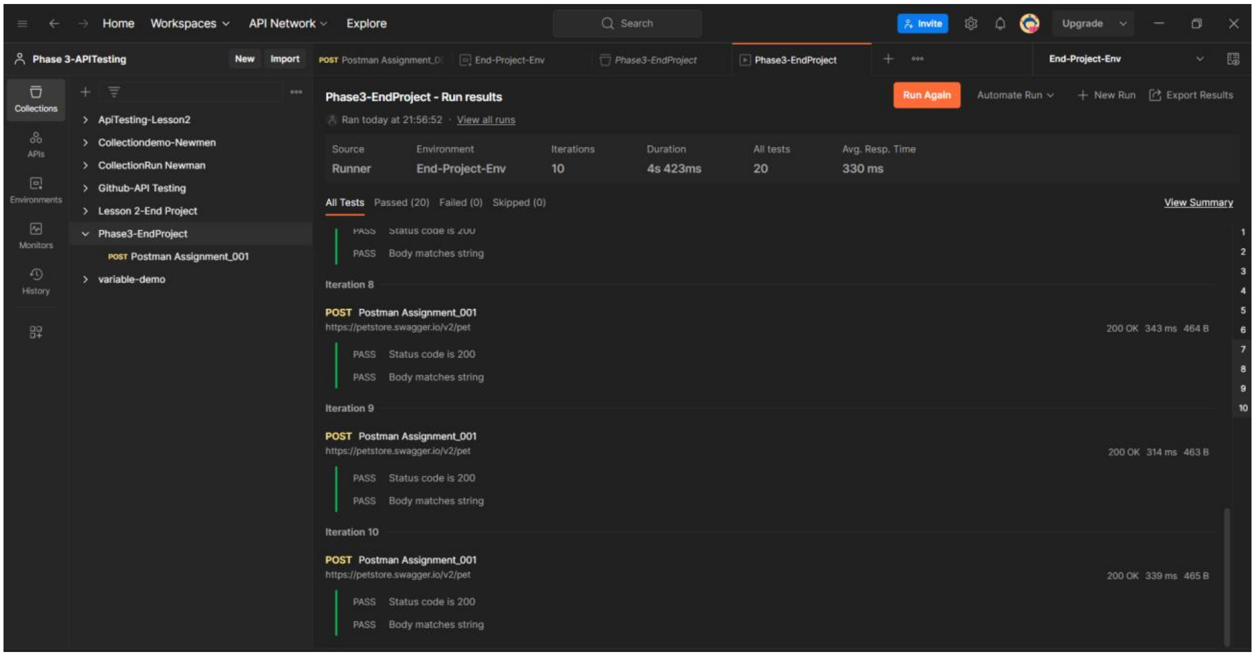
Iterations: 1

Delay: 0 ms

Data: Select File

□ Persist responses for a session

Above we have to click on select file and attach (projectdata.csv) then click on Run Phase 3-EndProject



Phase3-EndProject - Run results

Ran today at 21:56:52 · View all runs

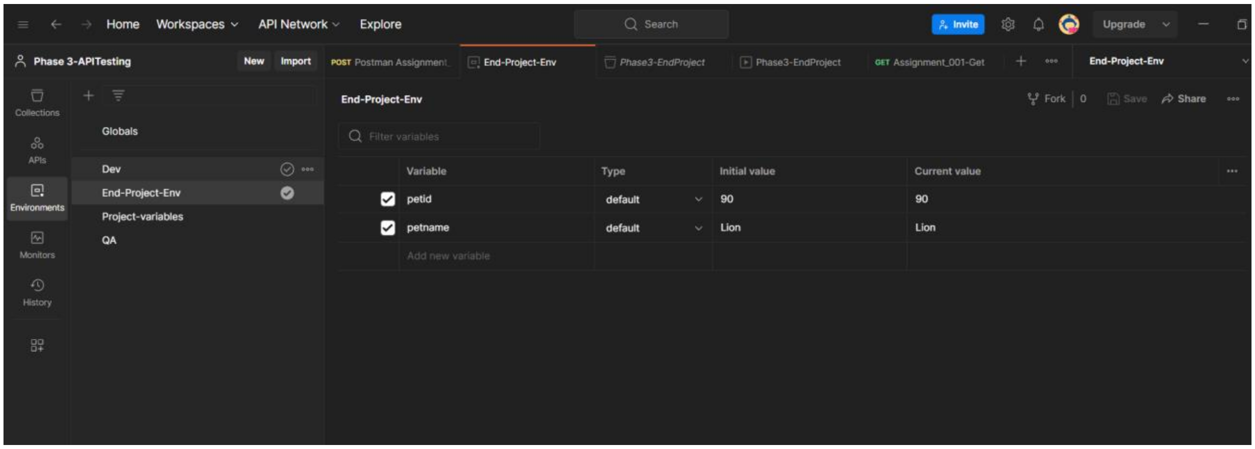
Source	Environment	Iterations	Duration	All tests	Avg. Resp. Time
Runner	End-Project-Env	10	4s 423ms	20	330 ms

All Tests Passed (20) Failed (0) Skipped (0)

```
    PASS   Status code is 200
    PASS   Body matches string

Iteration 8

POST  Postman Assignment_001
https://petstore.swagger.io/v2/pet                                200 OK   343 ms   464 B
    PASS   Status code is 200
    PASS   Body matches string

Iteration 9

POST  Postman Assignment_001
https://petstore.swagger.io/v2/pet                                200 OK   314 ms   463 B
    PASS   Status code is 200
    PASS   Body matches string

Iteration 10

POST  Postman Assignment_001
https://petstore.swagger.io/v2/pet                                200 OK   339 ms   465 B
    PASS   Status code is 200
    PASS   Body matches string
```

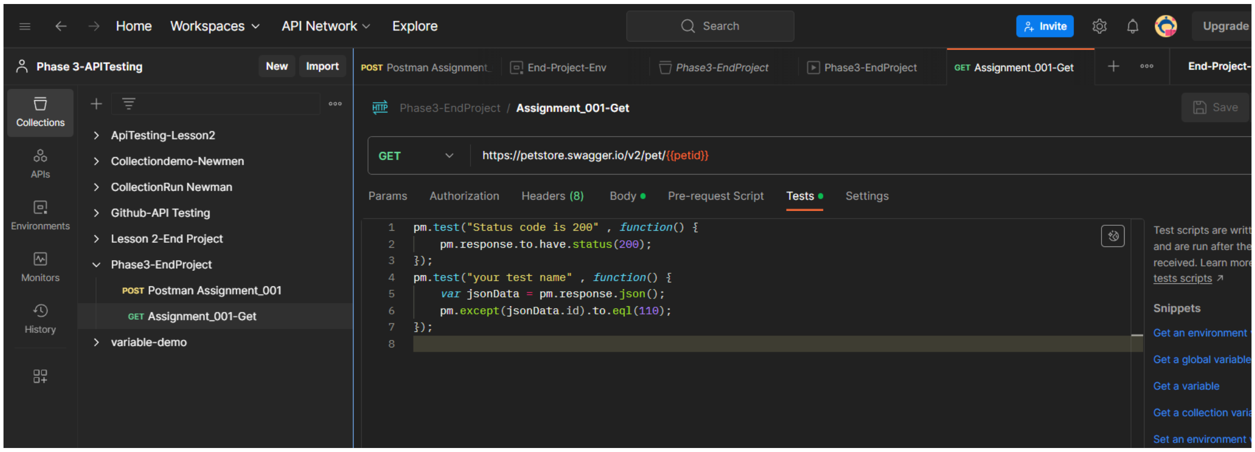
================

Validate petID: (Postman-Assignment001-get Method)



End-Project-Env

	Variable	Type	Initial value	Current value
☑	petid	default	90	90
☑	petname	default	Lion	Lion

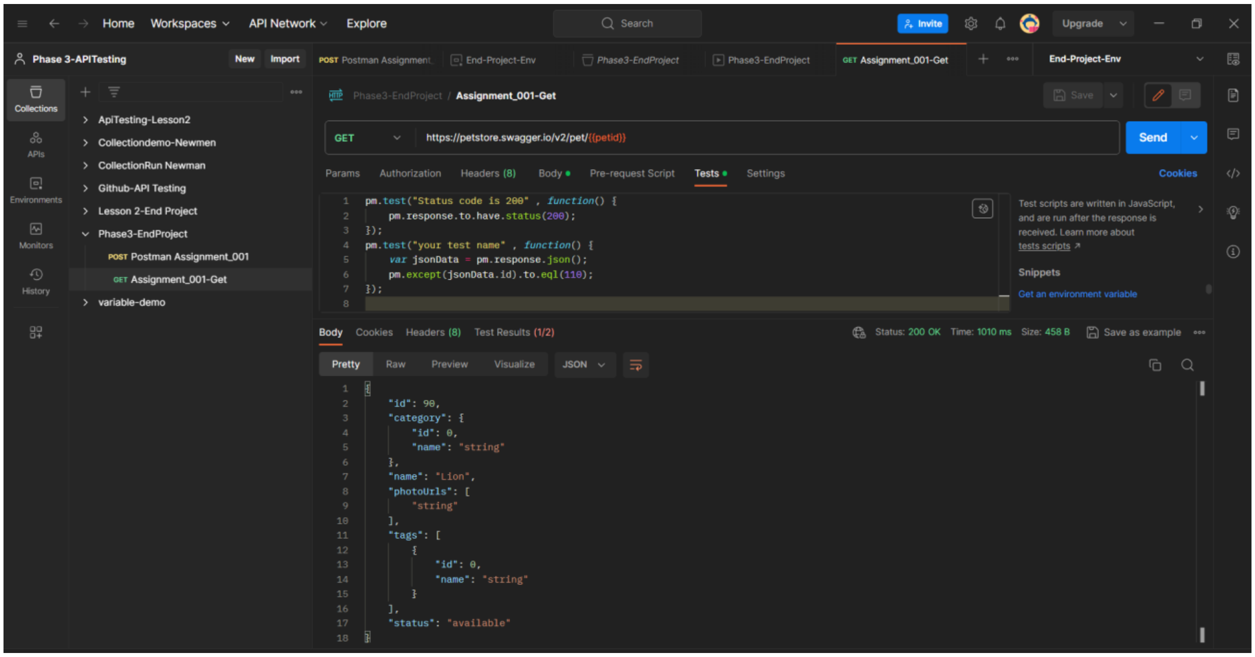


Phase3-EndProject / Assignment_001-Get

GET https://petstore.swagger.io/v2/pet/{{petid}}

```
pm.test("Status code is 200" , function() {
    pm.response.to.have.status(200);
});
pm.test("your test name" , function() {
    var jsonData = pm.response.json();
    pm.except(jsonData.id).to.eql(110);
});
```

Test scripts are writ and are run after the received. Learn more tests scripts



Phase3-EndProject / Assignment_001-Get

GET https://petstore.swagger.io/v2/pet/{{petid}}

```
pm.test("Status code is 200" , function() {
    pm.response.to.have.status(200);
});
pm.test("your test name" , function() {
    var jsonData = pm.response.json();
    pm.except(jsonData.id).to.eql(110);
});
```

Test scripts are written in JavaScript, and are run after the response is received. Learn more about tests scripts

Test Results (1/2)

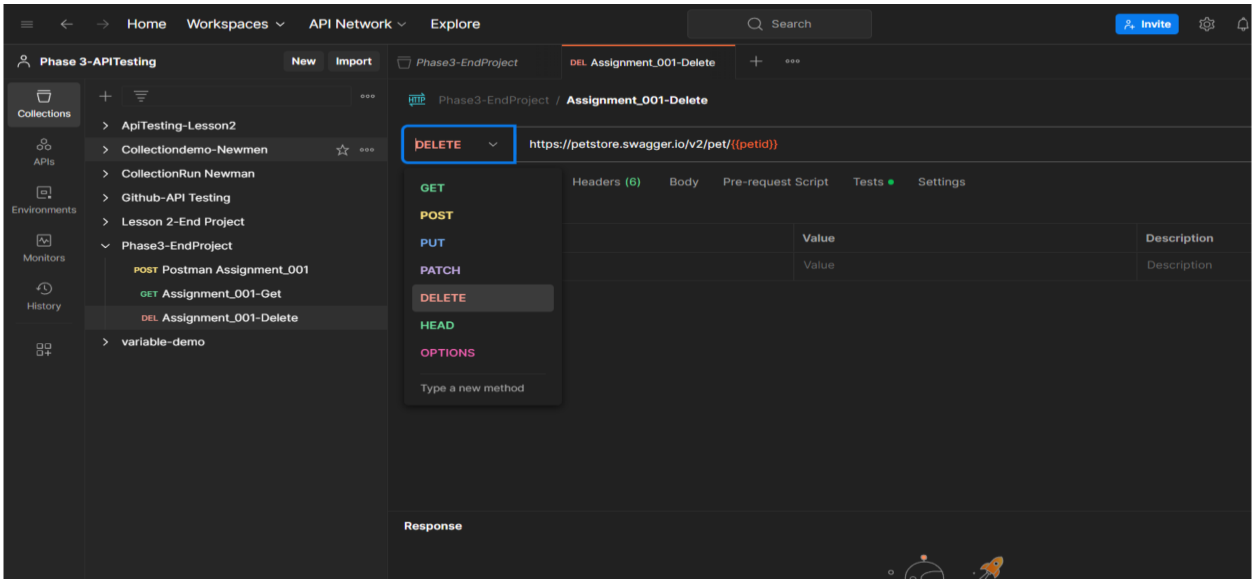
Status: 200 OK Time: 1010 ms Size: 458 B

```
{
    "id": 90,
    "category": {
        "id": 0,
        "name": "string"
    },
    "name": "Lion",
    "photoUrls": [
        "string"
    ],
    "tags": [
        {
            "id": 0,
            "name": "string"
        }
    ],
    "status": "available"
}
```

========================

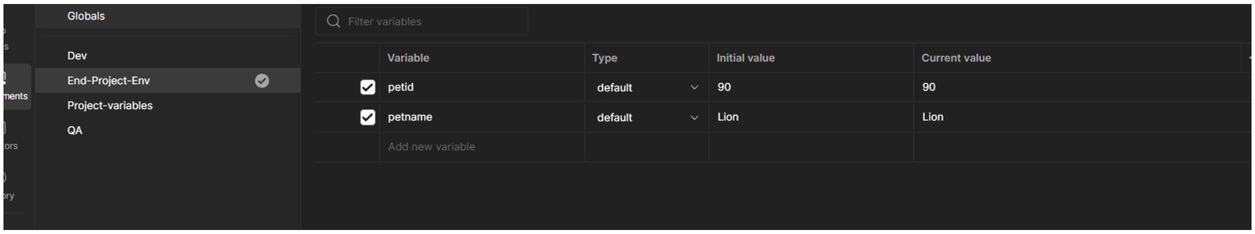
Delete petID: (Postman-Assignment001-delete Method)

To Add delete Request we must Add the request and then click next to the URL and the select delete as shown below.

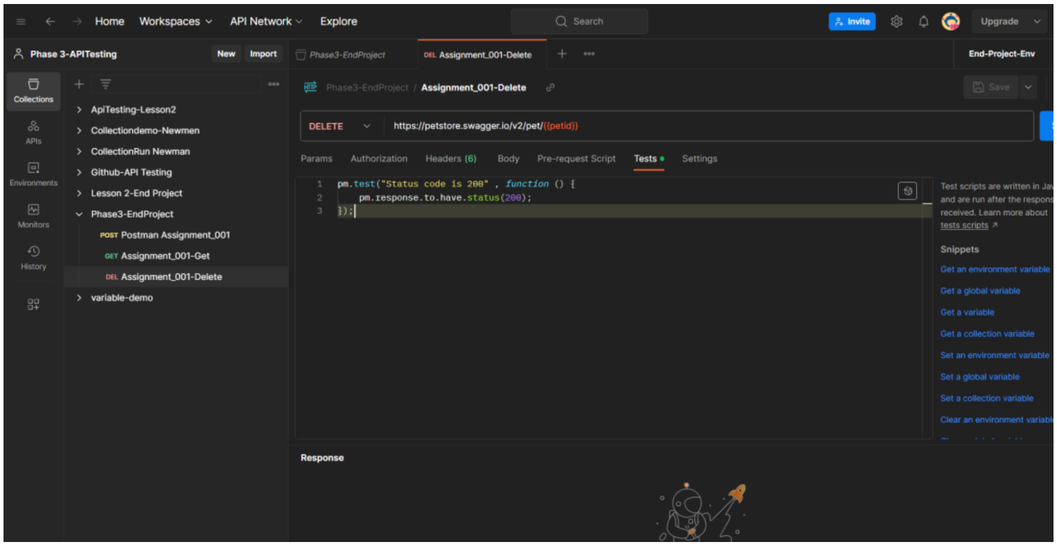


DELETE https://petstore.swagger.io/v2/pet/{{petid}}

Then add the URL as given https://petstore.swagger.io/v2/pet/{{petid}}
In this Environment part select End-Project-Env enter petid and petname



	Variable	Type	Initial value	Current value
☑	petid	default	90	90
☑	petname	default	Lion	Lion
	Add new variable			

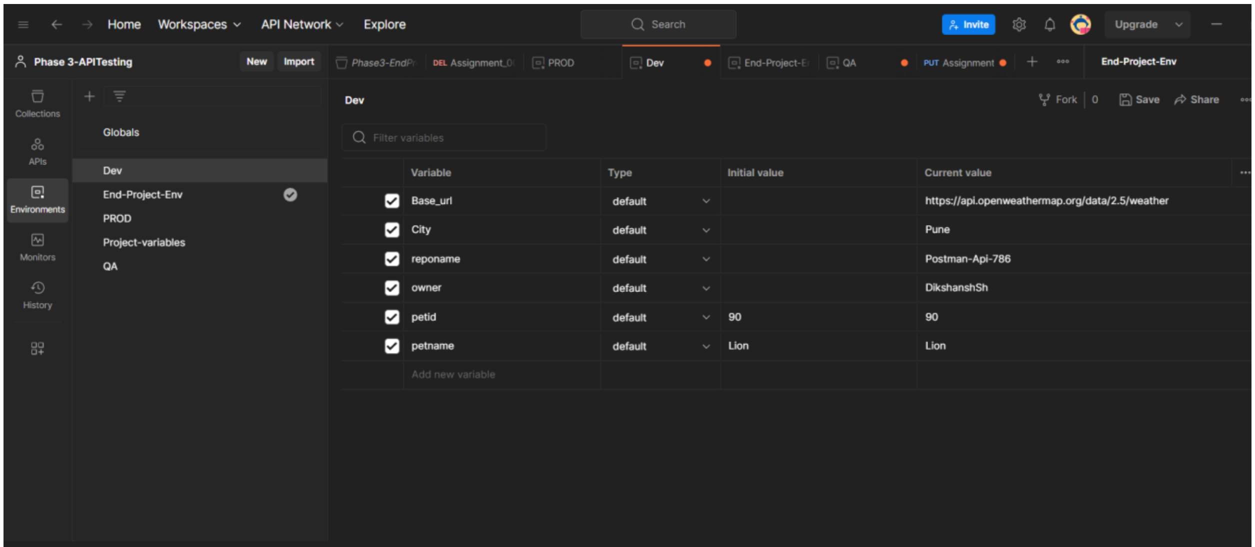


DELETE https://petstore.swagger.io/v2/pet/{{petid}}

```
pm.test("Status code is 200" , function () {
    pm.response.to.have.status(200);
});
```

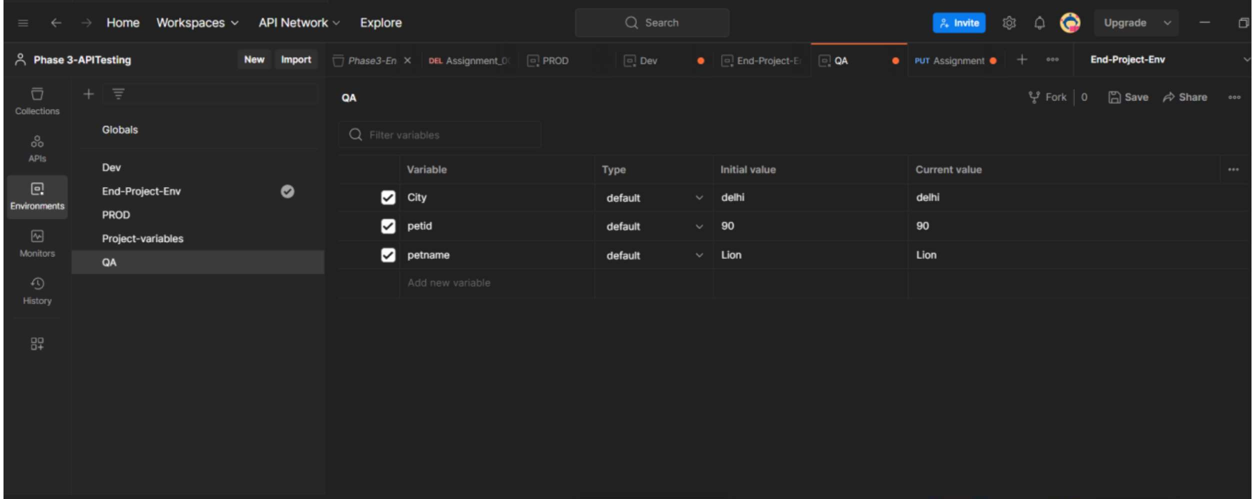
Postman Assignment_002:

Creating 3 test Environment as DEV, QA, PROD



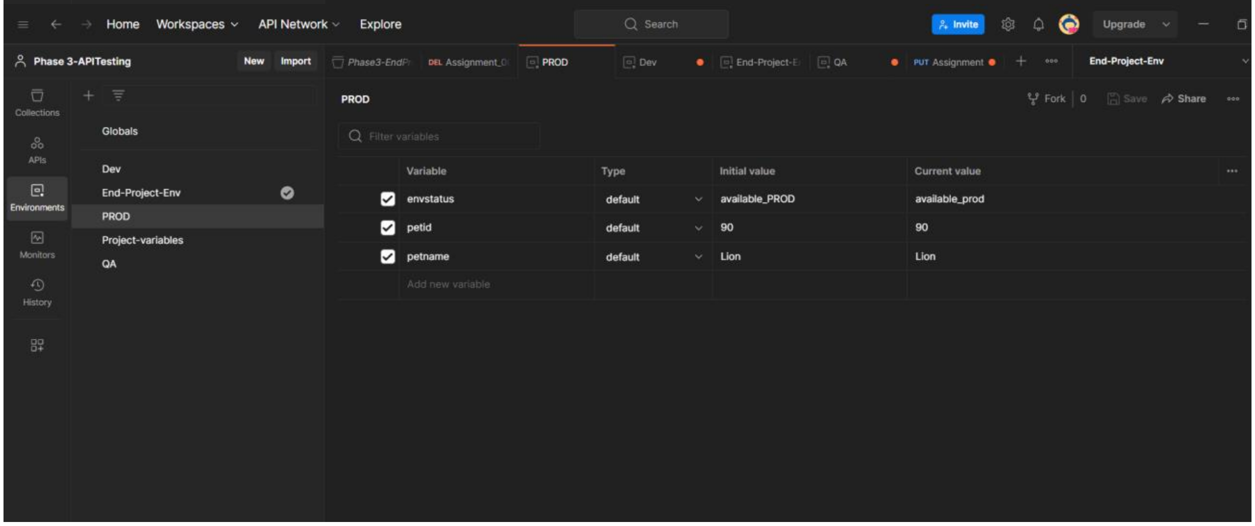
Dev

	Variable	Type	Initial value	Current value
☑	Base_url	default		https://api.openweathermap.org/data/2.5/weather
☑	City	default		Pune
☑	reponame	default		Postman-Api-786
☑	owner	default		DikshanshSh
☑	petid	default	90	90
☑	petname	default	Lion	Lion
	Add new variable			



QA

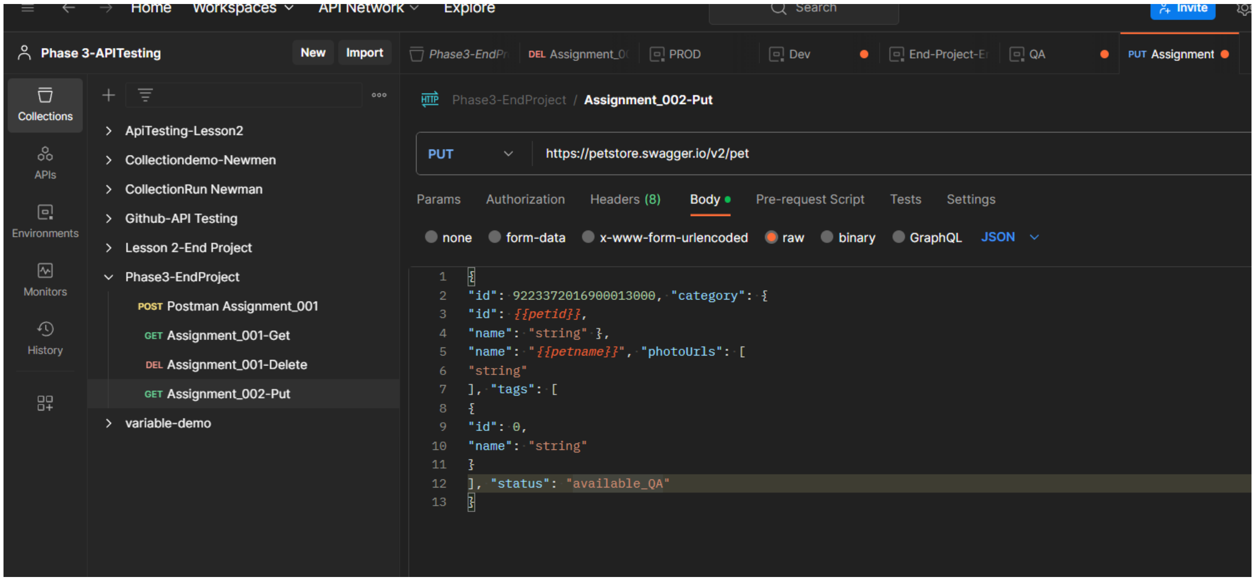
	Variable	Type	Initial value	Current value
☑	City	default	delhi	delhi
☑	petid	default	90	90
☑	petname	default	Lion	Lion
	Add new variable			



PROD

	Variable	Type	Initial value	Current value
☑	envstatus	default	available_PROD	available_prod
☑	petid	default	90	90
☑	petname	default	Lion	Lion
	Add new variable			

PUT Call request JSON body:

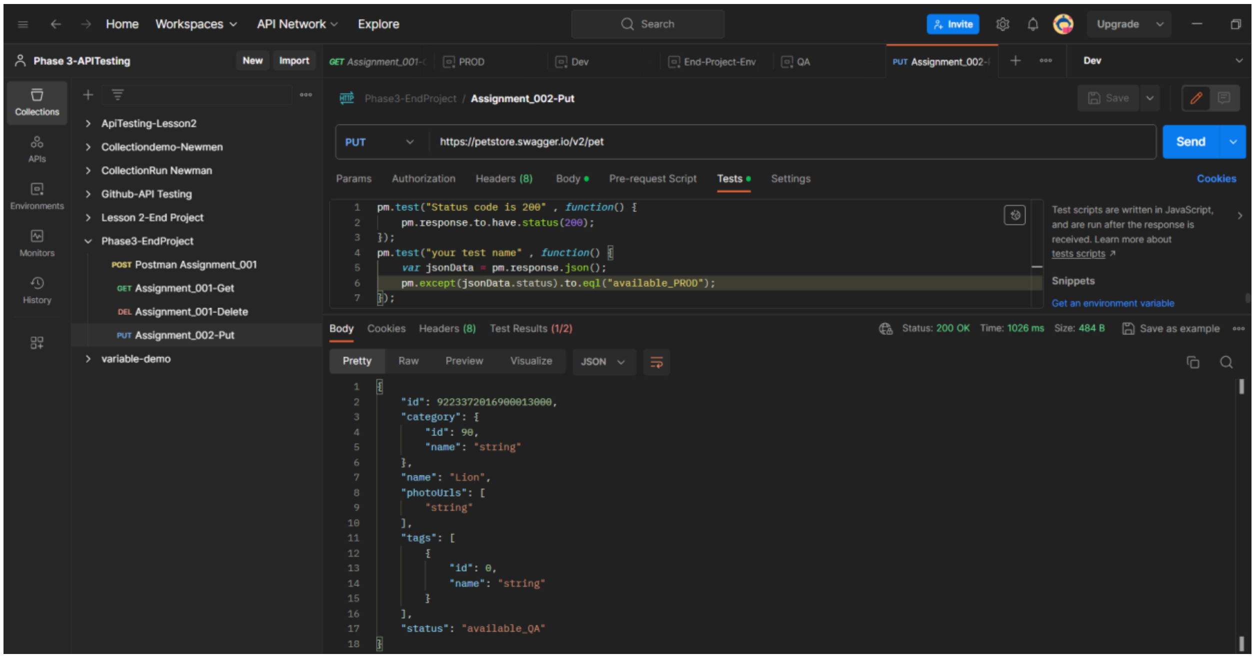


PUT https://petstore.swagger.io/v2/pet

○ none ○ form-data ○ x-www-form-urlencoded ● raw ○ binary ○ GraphQL JSON

```
{
"id": 9223372016900013000, "category": {
"id": {{petid}},
"name": "string" },
"name": "{{petname}}", "photoUrls": [
"string"
], "tags": [
{
"id": 0,
"name": "string"
}
], "status": "available_QA"
}
```

Response:



PUT https://petstore.swagger.io/v2/pet

```
pm.test("Status code is 200" , function() {
    pm.response.to.have.status(200);
});
pm.test("your test name" , function() {
    var jsonData = pm.response.json();
    pm.except(jsonData.status).to.eql("available_PROD");
});
```

Test scripts are written in JavaScript, and are run after the response is received. Learn more about tests scripts ↗

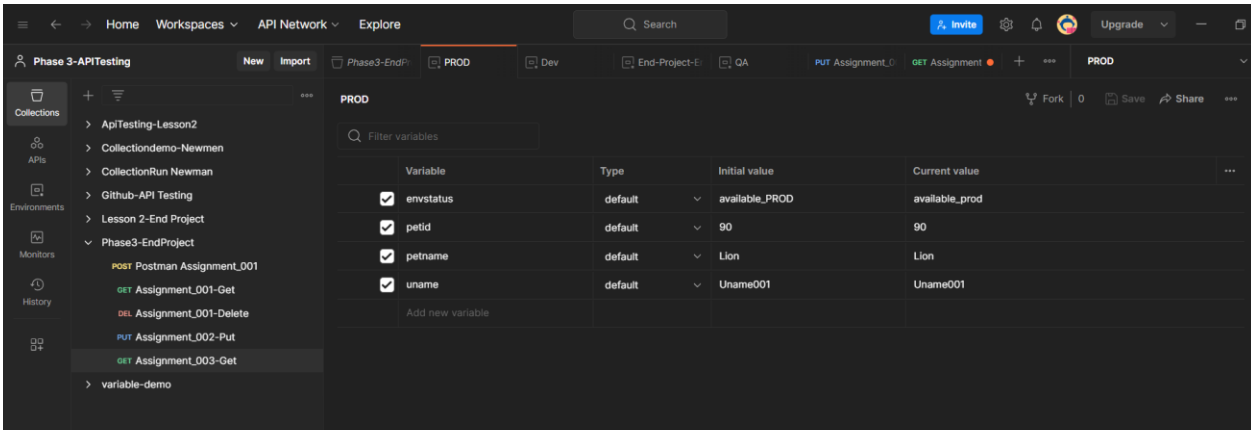
Status: 200 OK Time: 1026 ms Size: 484 B

```
{
    "id": 9223372016900013000,
    "category": {
        "id": 90,
        "name": "string"
    },
    "name": "Lion",
    "photoUrls": [
        "string"
    ],
    "tags": [
        {
            "id": 0,
            "name": "string"
        }
    ],
    "status": "available_QA"
}
```

====================

Postman Assignment_003:

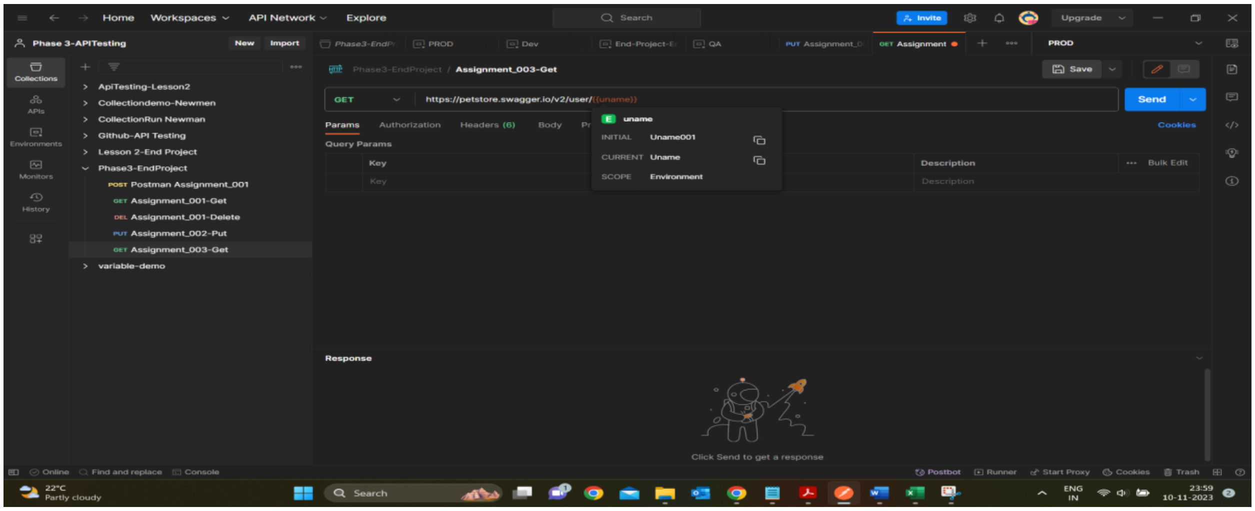
Creating a variable as uname with initial and current value as Uname001



PROD

	Variable	Type	Initial value	Current value
☑	envstatus	default	available_PROD	available_prod
☑	petid	default	90	90
☑	petname	default	Lion	Lion
☑	uname	default	Uname001	Uname001
	Add new variable			

Fetching a **GET** call with URL: https://petstore.swagger.io/v2/user/{{uname}} as shown below

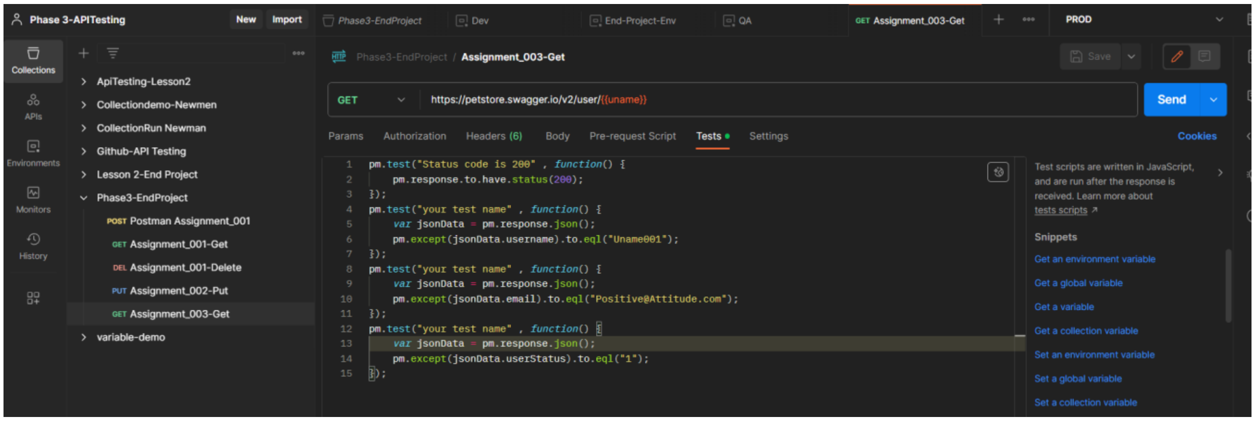


Phase3-EndProject / Assignment_003-Get

GET https://petstore.swagger.io/v2/user/{{uname}}

E uname
INITIAL Uname001
CURRENT Uname
SCOPE Environment

Click Send to get a response



Phase3-EndProject / Assignment_003-Get

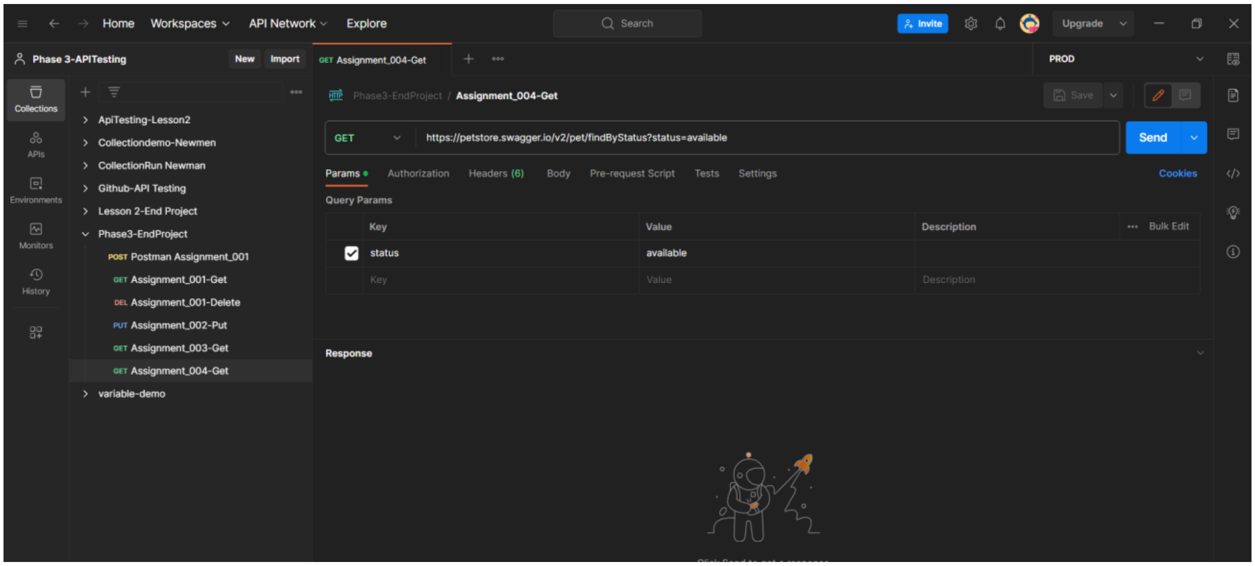
GET https://petstore.swagger.io/v2/user/{{uname}}

Tests

```
pm.test("Status code is 200" , function() {
    pm.response.to.have.status(200);
});
pm.test("your test name" , function() {
    var jsonData = pm.response.json();
    pm.except(jsonData.username).to.eql("Uname001");
});
pm.test("your test name" , function() {
    var jsonData = pm.response.json();
    pm.except(jsonData.email).to.eql("Positive@Attitude.com");
});
pm.test("your test name" , function() {
    var jsonData = pm.response.json();
    pm.except(jsonData.userStatus).to.eql("1");
});
```

Test scripts are written in JavaScript, and are run after the response is received. Learn more about tests scripts ↗

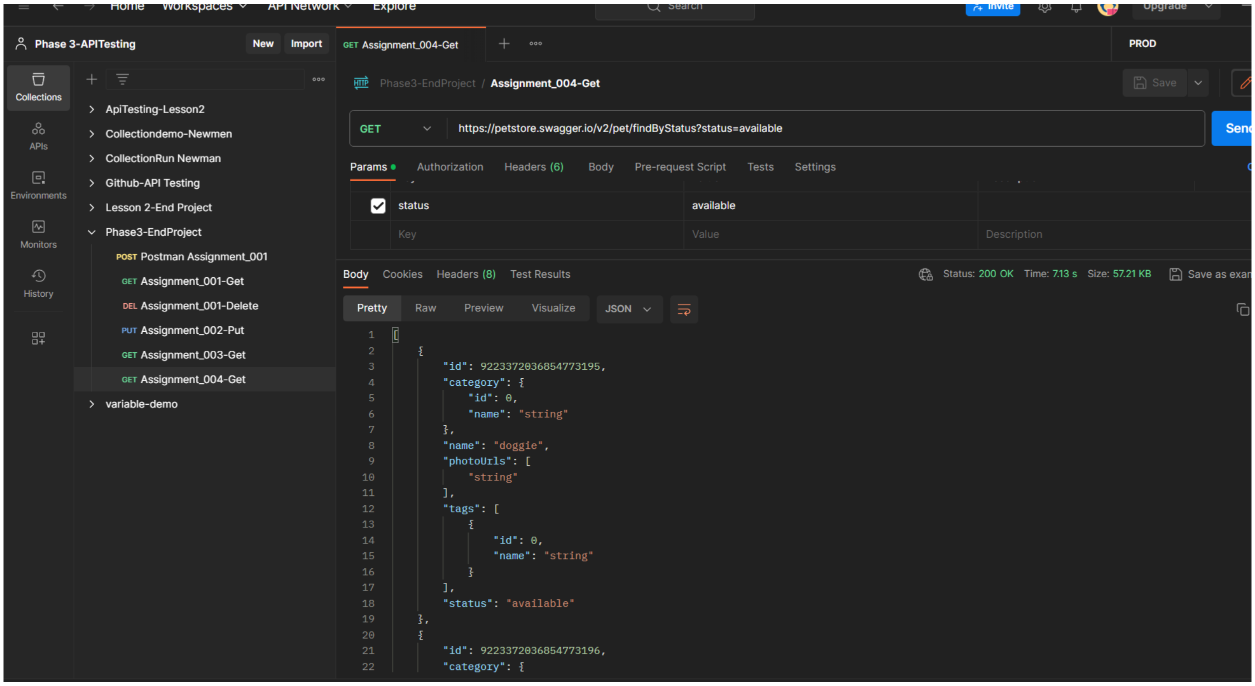
Postman Assignment_004:



GET https://petstore.swagger.io/v2/pet/findByStatus?status=available

	Key	Value	Description
☑	status	available	
	Key	Value	Description

Response:



GET https://petstore.swagger.io/v2/pet/findByStatus?status=available

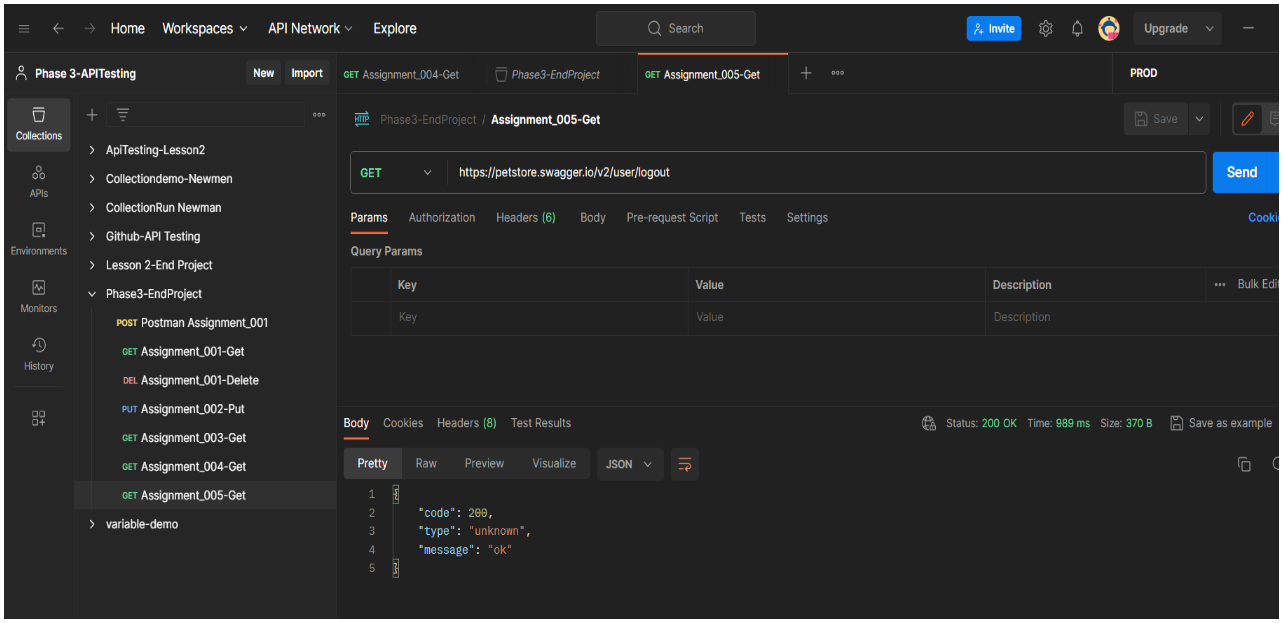
	Key	Value	Description
☑	status	available	
	Key	Value	Description

Status: 200 OK Time: 7.13 s Size: 57.21 KB

```
[
    {
        "id": 9223372036854773195,
        "category": {
            "id": 0,
            "name": "string"
        },
        "name": "doggie",
        "photoUrls": [
            "string"
        ],
        "tags": [
            {
                "id": 0,
                "name": "string"
            }
        ],
        "status": "available"
    },
    {
        "id": 9223372036854773196,
        "category": {
```

================

Postman Assignment_005:



GET https://petstore.swagger.io/v2/user/logout

Status: 200 OK Time: 989 ms Size: 370 B

```
{
    "code": 200,
    "type": "unknown",
    "message": "ok"
}
```

==

REST Assured Assignment_001 :

Go to Eclipse -> create project RestAssured -> create a package with name Course_EndProject -> Create a class Assignment1:

```
package Course_EndProject;

import java.io.File;

import org.apache.logging.log4j.LogManager;
import org.apache.logging.log4j.Logger;
import org.hamcrest.Matchers;
import org.testng.annotations.Test;

import io.restassured.RestAssured;
import io.restassured.http.ContentType;

public class Assignment1 {

Logger logger = LogManager.getLogger(Assignment1.class);

        @Test(priority='1')
        public void assignment1Post()
        {
                logger.info("Course_EndProject - Assignment1 - POST request");
```

```
            File file = new File("resources/datasheet.json");
            int id = RestAssured.given()
            .baseUri("https://petstore.swagger.io/v2/pet")
            .contentType(ContentType.JSON)
            .body(file)
            .when().post()
            .then()
            .statusCode(200)
            .log().all()
            .body("name", Matchers.equalTo("cat")).extract().path("id");
      logger.trace("The status code is checked");

      System.out.println(id);
      logger.trace("Id has been captured and validated");

      }

      @Test(priority='2',dependsOnMethods="assignment1Post")
      public void assignment1Get()
      {
            logger.info("Course_EndProject - Assignment1 - GET request");
      int id = RestAssured.given()
            .baseUri("https://petstore.swagger.io/v2/pet/452")
            .when()
            .get()
            .then()
            .statusCode(200)
            .log().all()
            .body("status", Matchers.equalTo("available")).extract().path("category.id");
   System.out.println(id); // 0
   logger.trace("Id and status has been captured and validated");
      }
      @Test(priority='3',dependsOnMethods="assignment1Get")
      public void assignment1Delete()
      {
            logger.info("Course_EndProject - Assignment1 - Delete request");
RestAssured.given()
            .baseUri("https://petstore.swagger.io/v2/pet/452")
            .when()
            .delete()
            .then()
            .statusCode(200)
            .log().all()
            .body("code", Matchers.equalTo(200))
            .body("type", Matchers.equalTo("unknown"))
            .body("message", Matchers.equalTo("452")      }}
```

REST Assured Assignment_002 :

```
package Course_EndProject;

import java.io.File;

import org.apache.logging.log4j.LogManager;
import org.apache.logging.log4j.Logger;
import org.hamcrest.Matchers;
import org.testng.annotations.Test;

import io.restassured.RestAssured;
import io.restassured.http.ContentType;

public class Assignment2 {

Logger logger = LogManager.getLogger(Assignment2.class);

        @Test(priority='1')
        public void assignment2Post()
        {
                logger.info("Course_EndProject - Assignment2 - POST request");
                File file = new File("resources/datasheet.json");
                int id = RestAssured.given()
                .baseUri("https://petstore.swagger.io/v2/pet")
                .contentType(ContentType.JSON)
                .body(file)
                .when().post()
                .then()
                .statusCode(200)
                .log().all()
                .body("name", Matchers.equalTo("cat")).extract().path("id");
        logger.trace("The status code is checked");

        System.out.println(id);

        logger.trace("Id has been captured and validated");

        }


        @Test(priority='2',dependsOnMethods = "assignment2Post")
        public void assignment002Put()
        {
                File file = new File("resources/dataput.json");
        int id = RestAssured.given()
                .baseUri("https://petstore.swagger.io/v2/pet/")
                .contentType(ContentType.JSON)
                .body(file)
                .when()
                .post()
                .then()
                .statusCode(200)
                .log().all()
                .body("status", Matchers.equalTo("available_QA")).extract().path("id");

        System.out.println(id);


        }
}
```

===

Rest Assured Assignment_003:

```
package Course_EndProject;

import org.hamcrest.Matchers;
import org.testng.annotations.Test;

import io.restassured.RestAssured;


public class Assignment3 {

        @Test
        public void assignment3User()
        {
                RestAssured.given()
                .baseUri("https://petstore.swagger.io/v2/user/Uname001")
                .when()
                .get()
                .then()
                .statusCode(200)
                .log().all()
                .body("username", Matchers.equalTo("Uname001"))
                .body("email", Matchers.equalTo("Positive@Attitude.com"))
                .body("userStatus", Matchers.equalTo(1));



        }
}
```

===

Rest Assured Assignment_004:

```
package Course_EndProject;

import org.hamcrest.Matchers;
import org.testng.annotations.Test;

import io.restassured.RestAssured;

public class Asssignment4 {

        @Test
        public void assignment004login()
        {
                RestAssured.given()
                .baseUri("https://petstore.swagger.io/v2/user/login")
                .auth().preemptive().basic("Uname001", "@tt!tude")
                .when()
                .get()
                .then()
                .statusCode(200)
                .log().all()
                .body("message", Matchers.anything("logged in user session"));

        }

}
```

==

Rest Assured Assignment_005:

```
package Course_EndProject;

import org.testng.annotations.Test;

import io.restassured.RestAssured;

public class Assignment5 {

        @Test
        public void assignment5FindStatus()
        {
                RestAssured.given()
                .baseUri("https://petstore.swagger.io/v2/pet/findByStatus")
                .queryParam("status", "available")
                .queryParam("status", "pending")
                .queryParam("status", "sold")
                .when()
                .get()
                .then()
                .statusCode(200)
                .log().all();

        }

}
```

===

Rest Assured Assignment_006:

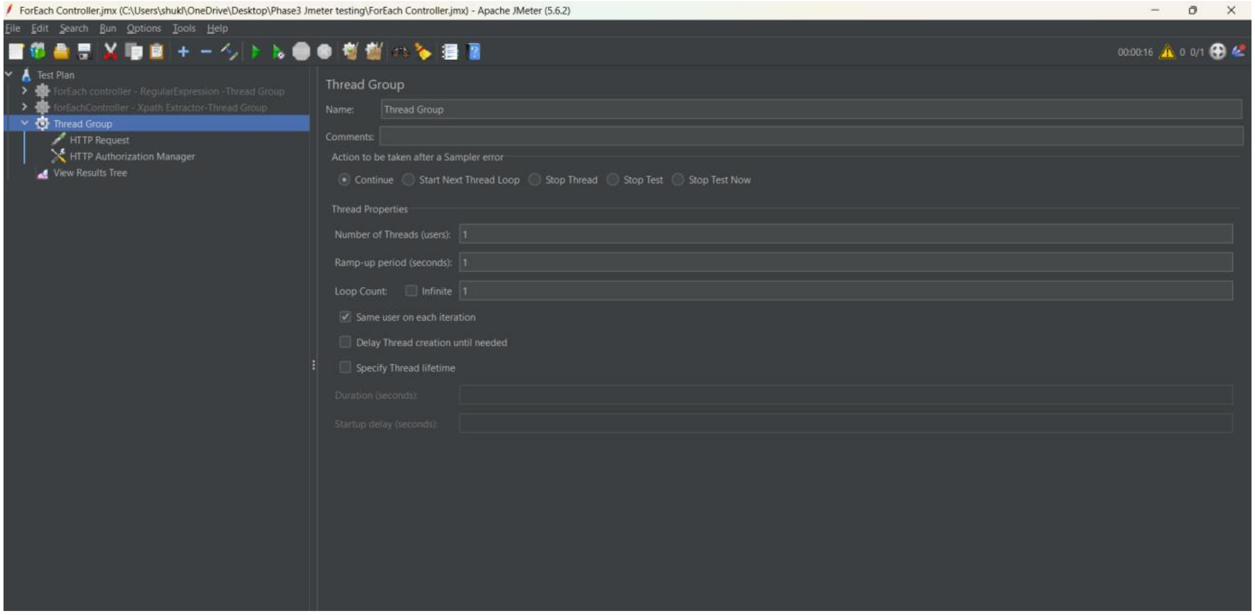
```
package Course_EndProject;

import org.hamcrest.Matchers;
import org.testng.annotations.Test;

import io.restassured.RestAssured;

public class Assignment6 {

        @Test
        public void assignment6Logout()
        {
                RestAssured.given()
                .baseUri("https://petstore.swagger.io/v2/user/logout")
                .when()
                .get()
                .then()
                .statusCode(200)
                .log().all()
                .body("code", Matchers.equalTo(200))
                .body("type", Matchers.equalTo("unknown"))
                .body("message", Matchers.equalTo("ok"));

        }


}
```

===

Jmeter Assignment:

- Validate https://httpbin.org/basic-auth/user/passwd link using HTTP Authentication manager
- Default username is **user**, and the default password is **passwd**

Add Thread Group



ForEach Controller.jmx (C:\Users\shukl\OneDrive\Desktop\Phase3 Jmeter testing\ForEach Controller.jmx) - Apache JMeter (5.6.2)
File Edit Search Run Options Tools Help
00:00:16 0 0/1
Test Plan
ForEach controller - RegularExpression -Thread Group
forEachController - Xpath Extractor-Thread Group
Thread Group
HTTP Request
HTTP Authorization Manager
View Results Tree

Thread Group

Name: Thread Group

Comments:

Action to be taken after a Sampler error

● Continue ○ Start Next Thread Loop ○ Stop Thread ○ Stop Test ○ Stop Test Now

Thread Properties

Number of Threads (users): 1

Ramp-up period (seconds): 1

Loop Count: □ Infinite 1

☑ Same user on each iteration

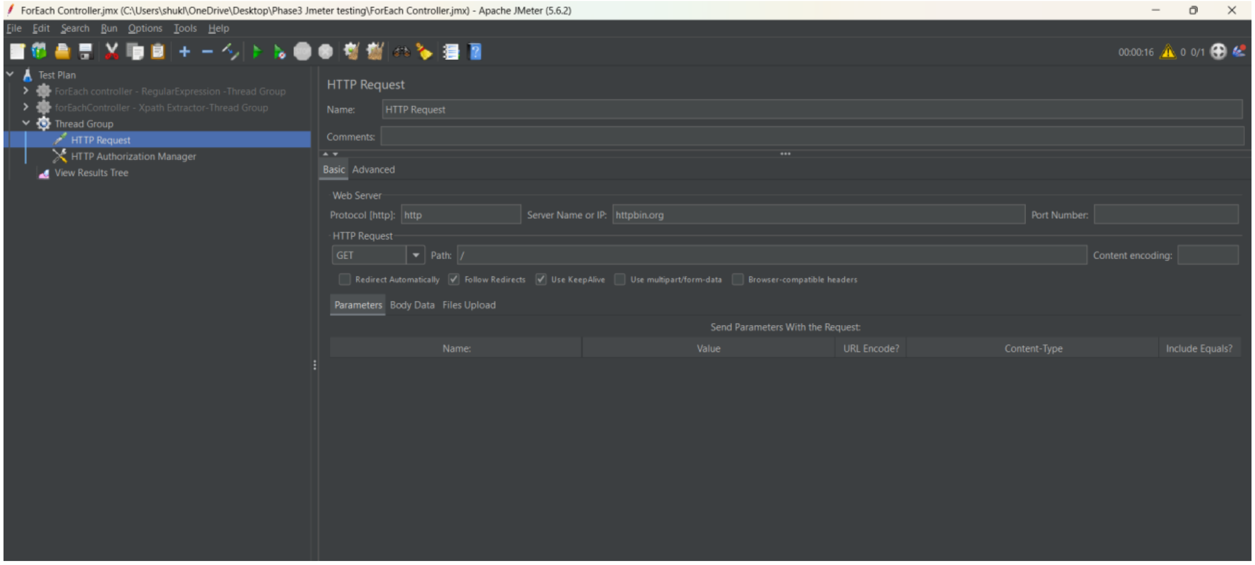
□ Delay Thread creation until needed

□ Specify Thread lifetime

Duration (seconds):

Startup delay (seconds):

Right click on thread group -> Add -> Sampler -> create HTTP Request



ForEach Controller.jmx (C:\Users\shukl\OneDrive\Desktop\Phase3 Jmeter testing\ForEach Controller.jmx) - Apache JMeter (5.6.2)
File Edit Search Run Options Tools Help
00:00:16 0 0/1
Test Plan
ForEach controller - RegularExpression -Thread Group
forEachController - Xpath Extractor-Thread Group
Thread Group
HTTP Request
HTTP Authorization Manager
View Results Tree
Basic Advanced
Parameters Body Data Files Upload

HTTP Request

Name: HTTP Request

Comments:

Web Server

Protocol [http]: http Server Name or IP: httpbin.org Port Number:

HTTP Request

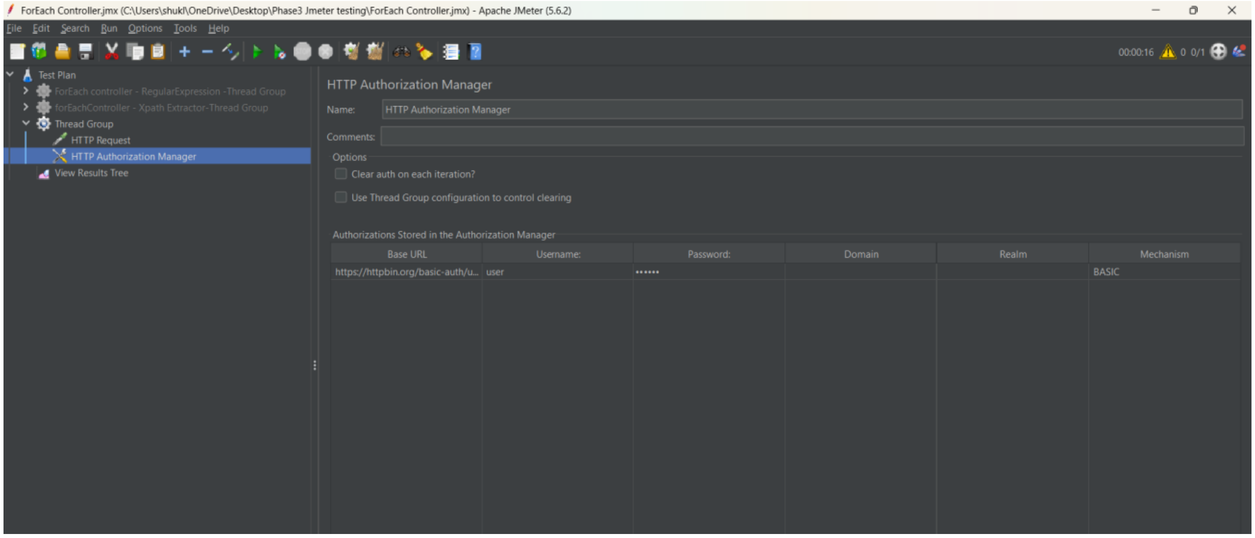
GET Path: / Content encoding:

□ Redirect Automatically ☑ Follow Redirects ☑ Use KeepAlive □ Use multipart/form-data □ Browser-compatible headers

Send Parameters With the Request:

Name:	Value	URL Encode?	Content-Type	Include Equals?

Right click on thread group -> Add -> Config Element -> HTTP Authorization Manager


ForEach Controller.jmx (C:\Users\shukl\OneDrive\Desktop\Phase3 Jmeter testing\ForEach Controller.jmx) - Apache JMeter (5.6.2)
File Edit Search Run Options Tools Help
00:00:16 0 0/1
Test Plan
ForEach controller - RegularExpression -Thread Group
forEachController - Xpath Extractor-Thread Group
Thread Group
HTTP Request
HTTP Authorization Manager
View Results Tree

HTTP Authorization Manager

Name: HTTP Authorization Manager

Comments:

Options

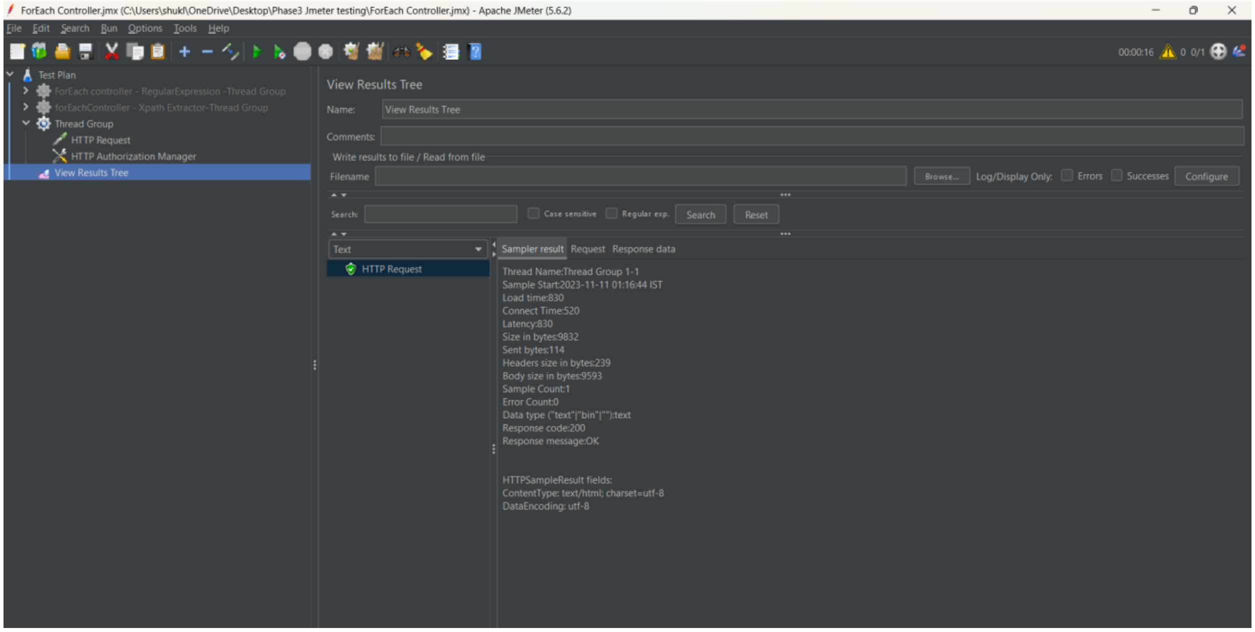
□ Clear auth on each iteration?

□ Use Thread Group configuration to control clearing

Authorizations Stored in the Authorization Manager

Base URL	Username:	Password:	Domain	Realm	Mechanism
https://httpbin.org/basic-auth/u...	user	••••••			BASIC

Right click on thread group -> Listener -> View Result Tree


ForEach Controller.jmx (C:\Users\shukl\OneDrive\Desktop\Phase3 Jmeter testing\ForEach Controller.jmx) - Apache JMeter (5.6.2)
File Edit Search Run Options Tools Help
00:00:16 0 0/1
Test Plan
ForEach controller - RegularExpression -Thread Group
forEachController - Xpath Extractor-Thread Group
Thread Group
HTTP Request
HTTP Authorization Manager
View Results Tree
Browse...
Search
Reset
Configure
Text
Sampler result
Request
Response data

View Results Tree

Name: View Results Tree

Comments:

Write results to file / Read from file

Filename Log/Display Only: □ Errors □ Successes

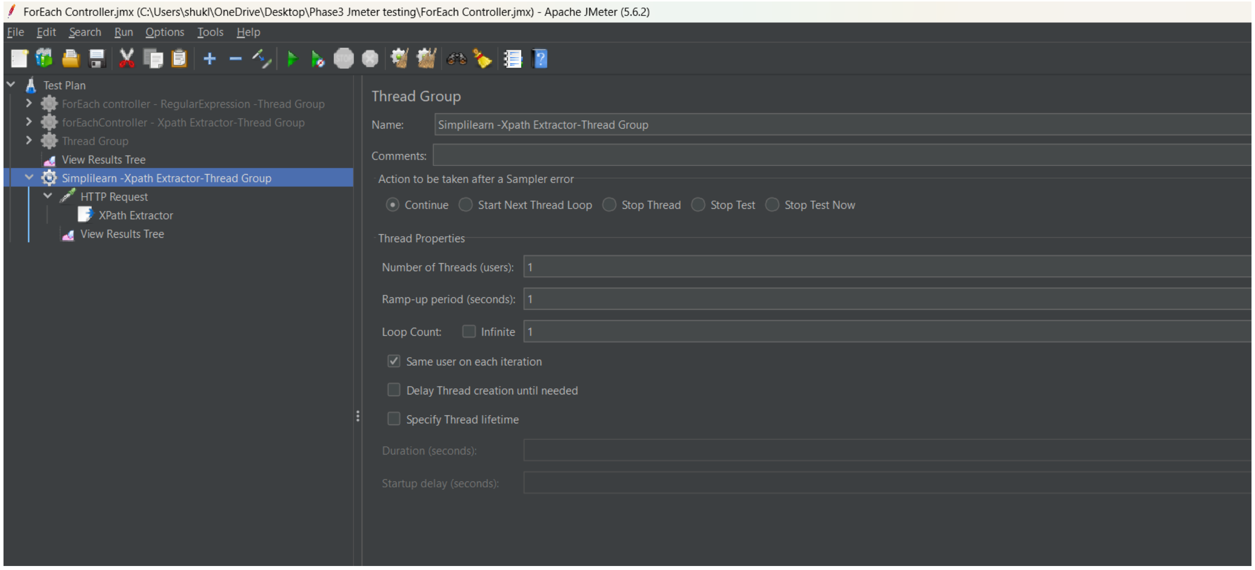
Search: □ Case sensitive □ Regular exp.

HTTP Request

```
Thread Name:Thread Group 1-1
Sample Start:2023-11-11 01:16:44 IST
Load time:830
Connect Time:520
Latency:830
Size in bytes:9832
Sent bytes:114
Headers size in bytes:239
Body size in bytes:9593
Sample Count:1
Error Count:0
Data type ("text"|"bin"|""):text
Response code:200
Response message:OK


HTTPSampleResult fields:
ContentType: text/html; charset=utf-8
DataEncoding: utf-8
```

- **Using HTTP sampler, hit https://www.simplilearn.com and validate xpath as //img[@title='Simplilearn - Online Certification Training Course Provider']**

Create a thread group with name Simplilearn – Xpath Extractor- Thread Group as shown below:



ForEach Controller.jmx (C:\Users\shukl\OneDrive\Desktop\Phase3 Jmeter testing\ForEach Controller.jmx) - Apache JMeter (5.6.2)

File Edit Search Run Options Tools Help
Test Plan
ForEach controller - RegularExpression -Thread Group
forEachController - Xpath Extractor-Thread Group
Thread Group
View Results Tree
Simplilearn -Xpath Extractor-Thread Group
HTTP Request
XPath Extractor
View Results Tree

Thread Group

Name: Simplilearn -Xpath Extractor-Thread Group

Comments:

Action to be taken after a Sampler error

● Continue ○ Start Next Thread Loop ○ Stop Thread ○ Stop Test ○ Stop Test Now

Thread Properties

Number of Threads (users): 1

Ramp-up period (seconds): 1

Loop Count: □ Infinite 1

☑ Same user on each iteration

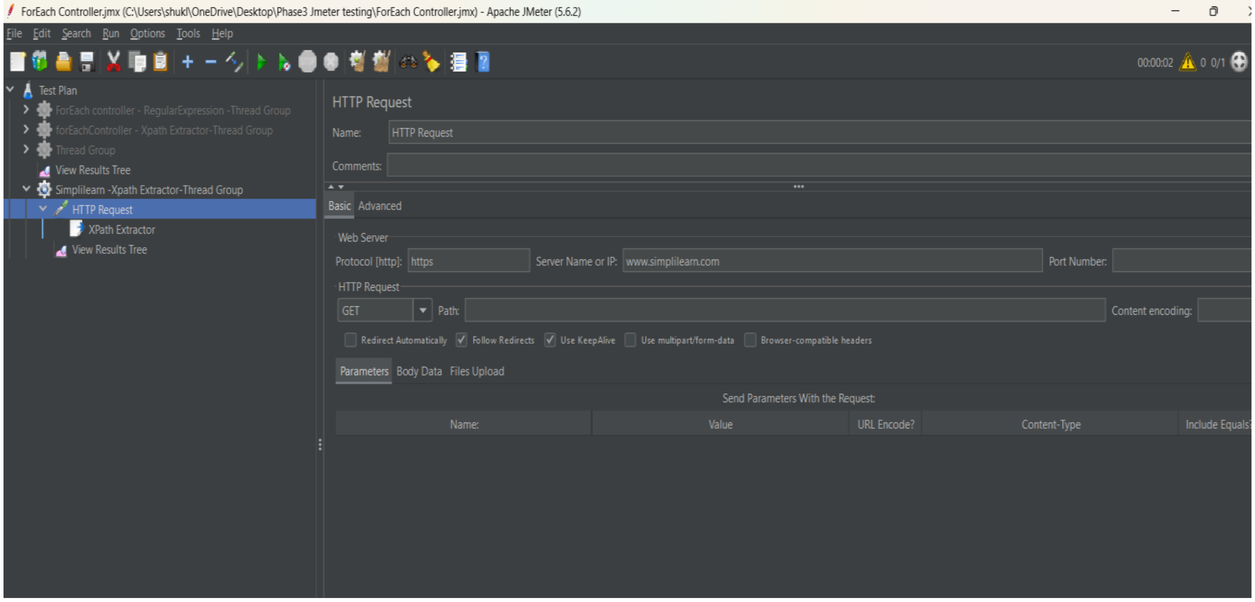
□ Delay Thread creation until needed

□ Specify Thread lifetime

Duration (seconds):

Startup delay (seconds):

Right click on thread group -> Add -> Sampler -> create HTTP Request



ForEach Controller.jmx (C:\Users\shukl\OneDrive\Desktop\Phase3 Jmeter testing\ForEach Controller.jmx) - Apache JMeter (5.6.2)

File Edit Search Run Options Tools Help
00:00:02 0 0/1
Test Plan
ForEach controller - RegularExpression -Thread Group
forEachController - Xpath Extractor-Thread Group
Thread Group
View Results Tree
Simplilearn -Xpath Extractor-Thread Group
HTTP Request
XPath Extractor
View Results Tree
Basic Advanced

HTTP Request

Name: HTTP Request

Comments:

Web Server

Protocol [http]: https Server Name or IP: www.simplilearn.com Port Number:

HTTP Request

GET Path: Content encoding:

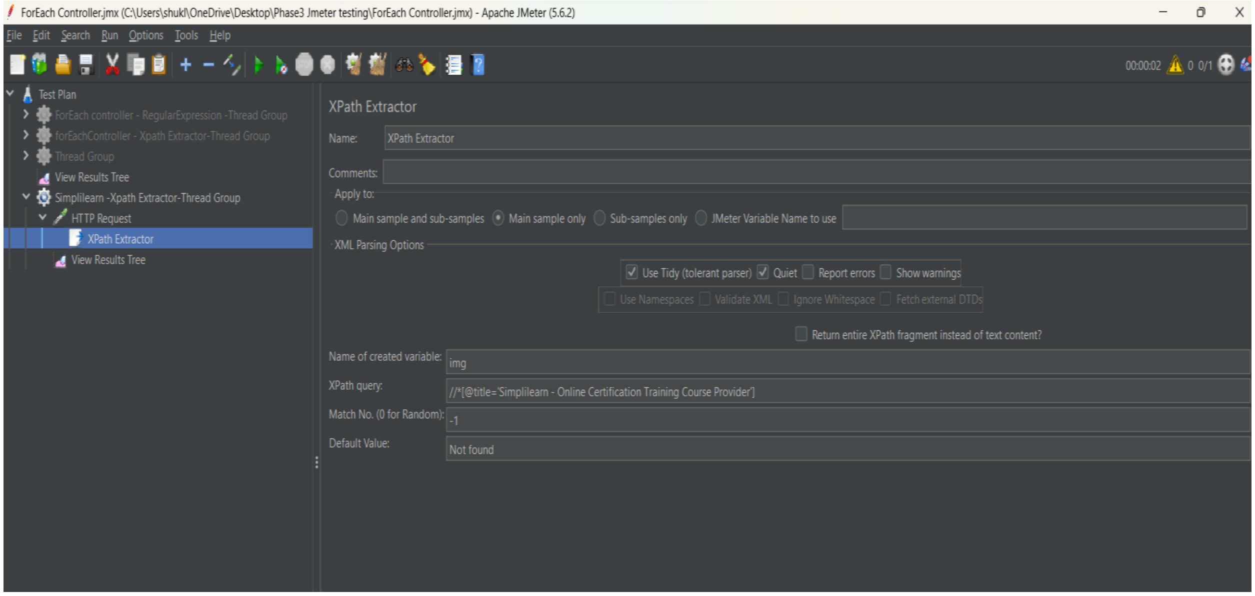
□ Redirect Automatically ☑ Follow Redirects ☑ Use KeepAlive □ Use multipart/form-data □ Browser-compatible headers

Parameters Body Data Files Upload

Send Parameters With the Request:

Name:	Value	URL Encode?	Content-Type	Include Equals?

Right click on HTTP Request -> Add -> Post processor -> XPath Extractor



ForEach Controller.jmx (C:\Users\shukl\OneDrive\Desktop\Phase3 Jmeter testing\ForEach Controller.jmx) - Apache JMeter (5.6.2)

- Test Plan
  - ForEach controller - RegularExpression -Thread Group
  - forEachController - Xpath Extractor-Thread Group
  - Thread Group
  - View Results Tree
  - Simplilearn -Xpath Extractor-Thread Group
    - HTTP Request
      - XPath Extractor
    - View Results Tree

XPath Extractor

Name: XPath Extractor

Comments:

Apply to:

○ Main sample and sub-samples ● Main sample only ○ Sub-samples only ○ JMeter Variable Name to use

XML Parsing Options

☑ Use Tidy (tolerant parser) ☑ Quiet □ Report errors □ Show warnings

□ Use Namespaces □ Validate XML □ Ignore Whitespace □ Fetch external DTDs

□ Return entire XPath fragment instead of text content?

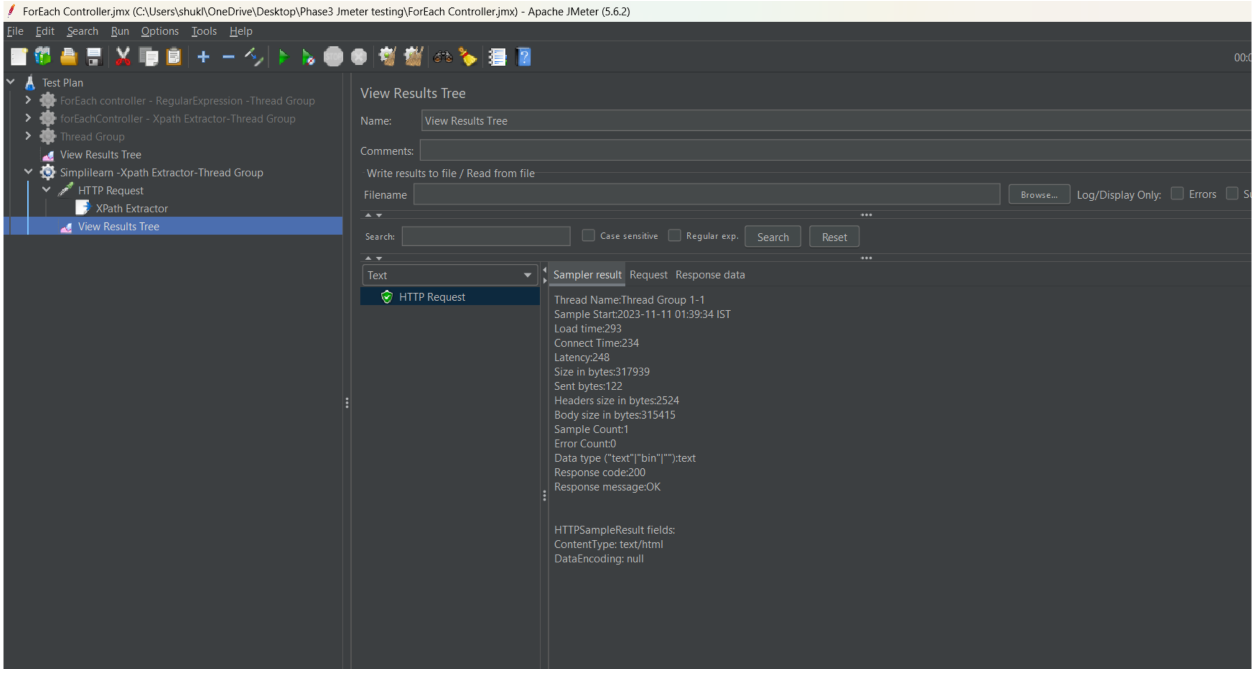
Name of created variable: img

XPath query: //*[@title='Simplilearn - Online Certification Training Course Provider']

Match No. (0 for Random): -1

Default Value: Not found

Result Response:



ForEach Controller.jmx (C:\Users\shukl\OneDrive\Desktop\Phase3 Jmeter testing\ForEach Controller.jmx) - Apache JMeter (5.6.2)

- Test Plan
  - ForEach controller - RegularExpression -Thread Group
  - forEachController - Xpath Extractor-Thread Group
  - Thread Group
  - View Results Tree
  - Simplilearn -Xpath Extractor-Thread Group
    - HTTP Request
      - XPath Extractor
    - View Results Tree

View Results Tree

Name: View Results Tree

Comments:

Write results to file / Read from file

Filename

Log/Display Only: □ Errors □ S

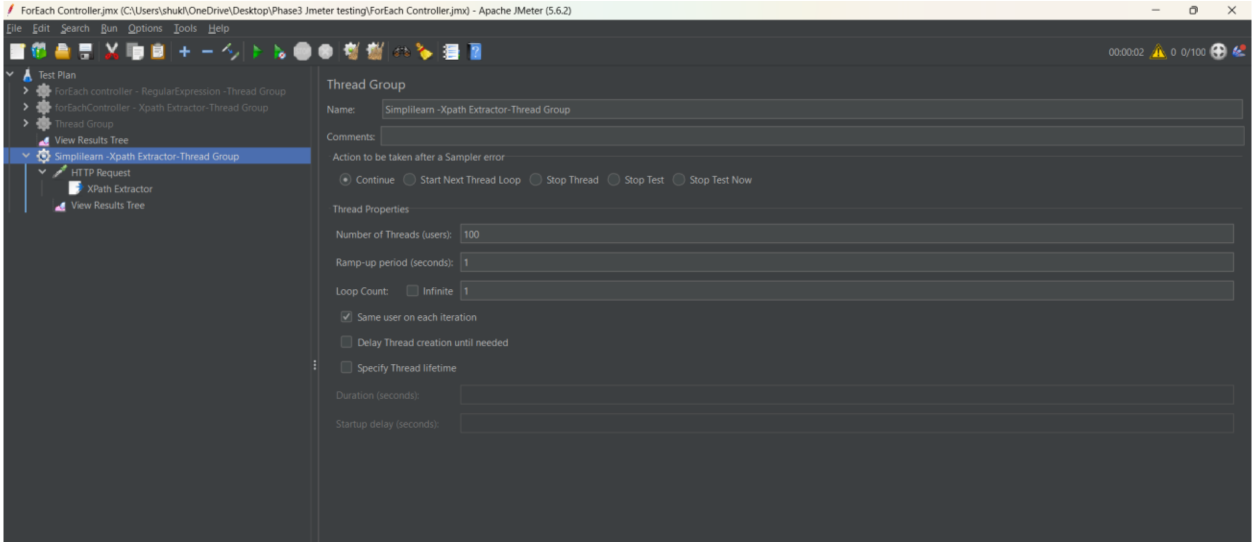
Search: □ Case sensitive □ Regular exp.

HTTP Request

```
Thread Name:Thread Group 1-1
Sample Start:2023-11-11 01:39:34 IST
Load time:293
Connect Time:234
Latency:248
Size in bytes:317939
Sent bytes:122
Headers size in bytes:2524
Body size in bytes:315415
Sample Count:1
Error Count:0
Data type ("text"|"bin"|""):text
Response code:200
Response message:OK


HTTPSampleResult fields:
ContentType: text/html
DataEncoding: null
```

===

- **Increase the thread count and do lead testing**



ForEach Controller.jmx (C:\Users\shukl\OneDrive\Desktop\Phase3 Jmeter testing\ForEach Controller.jmx) - Apache JMeter (5.6.2)

File Edit Search Run Options Tools Help
00:00:02 0 0/100
Test Plan
ForEach controller - RegularExpression -Thread Group
forEachController - Xpath Extractor-Thread Group
Thread Group
View Results Tree
Simplilearn -Xpath Extractor-Thread Group
HTTP Request
XPath Extractor
View Results Tree

Thread Group

Name: Simplilearn -Xpath Extractor-Thread Group

Comments:

Action to be taken after a Sampler error

● Continue ○ Start Next Thread Loop ○ Stop Thread ○ Stop Test ○ Stop Test Now

Thread Properties

Number of Threads (users): 100

Ramp-up period (seconds): 1

Loop Count: □ Infinite 1

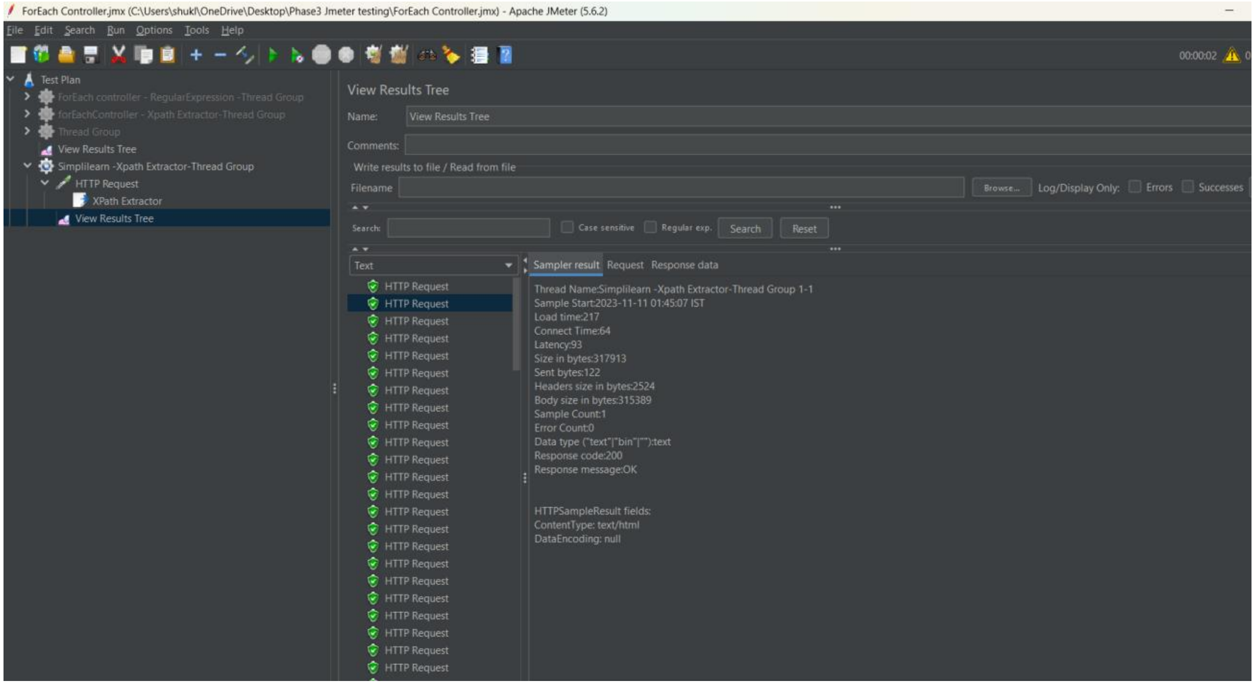
☑ Same user on each iteration

□ Delay Thread creation until needed

□ Specify Thread lifetime

Duration (seconds):

Startup delay (seconds):



ForEach Controller.jmx (C:\Users\shukl\OneDrive\Desktop\Phase3 Jmeter testing\ForEach Controller.jmx) - Apache JMeter (5.6.2)

File Edit Search Run Options Tools Help
00:00:02
Test Plan
ForEach controller - RegularExpression -Thread Group
forEachController - Xpath Extractor-Thread Group
Thread Group
View Results Tree
Simplilearn -Xpath Extractor-Thread Group
HTTP Request
XPath Extractor
View Results Tree

View Results Tree

Name: View Results Tree

Comments:

Write results to file / Read from file

Filename Browse... Log/Display Only: □ Errors □ Successes

Search: □ Case sensitive □ Regular exp. Search Reset

Text

Sampler result Request Response data

HTTP Request
HTTP Request
HTTP Request
HTTP Request
HTTP Request
HTTP Request
HTTP Request
HTTP Request
HTTP Request
HTTP Request
HTTP Request
HTTP Request
HTTP Request
HTTP Request
HTTP Request
HTTP Request
HTTP Request
HTTP Request
HTTP Request
HTTP Request
HTTP Request
HTTP Request
HTTP Request

```
Thread Name:Simplilearn -Xpath Extractor-Thread Group 1-1
Sample Start:2023-11-11 01:45:07 IST
Load time:217
Connect Time:64
Latency:93
Size in bytes:317913
Sent bytes:122
Headers size in bytes:2524
Body size in bytes:315389
Sample Count:1
Error Count:0
Data type ("text"|"bin"|""):text
Response code:200
Response message:OK


HTTPSampleResult fields:
ContentType: text/html
DataEncoding: null
```