

<https://github.com/naveenmpc/Phase3-Endproject-LessonEndProject>

<https://github.com/naveenmpc/JavaCodes1/tree/master/src>

Source Code:

Project Using rest assured, and Jmeter and Postman. .

```
package courseendproject1;

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 Assignment001 {

    Logger logger = LogManager.getLogger(Assignment001.class);

    @Test(priority='1')
    public void assignment001Post()
    {
        logger.info("Course End project - Assignment001 - POST request");
        File file = new File("resources/data.json");
        int id = RestAssured.given()
            .baseUrl("https://petstore.swagger.io/v2/pet")
            .contentType(ContentType.JSON)
            .body(file)
            .when().post()
            .then()
            .statusCode(200)
            .log().all()
            .body("name", Matchers.equalTo("duck")).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="assignment001Post")
    public void assignment001Get()
    {
        logger.info("Course End project - Assignment001 - GET request");
        int id = RestAssured.given()
            .baseUrl("https://petstore.swagger.io/v2/pet/987")
            .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="assignment001Get")
        public void assignment001Delete()
        {
            logger.info("Course End project - Assignment001 - Delete request");
            RestAssured.given()
                .baseUrl("https://petstore.swagger.io/v2/pet/987")
                .when()
                .delete()
                .then()
                .statusCode(200)
                .log().all()
                .body("code", Matchers.equalTo(200))
                .body("type", Matchers.equalTo("unknown"))
                .body("message", Matchers.equalTo("987"));

        }
    }
}

```

The screenshot shows an IDE console window with the following content:

```

<terminated> Assignment001 [TestNG] C:\Users\mnave\p2\pool\plugins\org.eclipse.justjopenjdk.hotspot.jre.full.win32.x86_64.17.0.8.v20230831-1047\jre\bin\javaw.exe (10-Nov-2023, 5:07:14 pm - 5:07:21 pm)
[INFO ] 2023-11-10 17:07:17.023 [main] Assignment001 - Course End project - Assignment001 - POST request
HTTP/1.1 200 OK
Date: Fri, 10 Nov 2023 11:37:17 GMT
Content-Type: application/json
Transfer-Encoding: chunked
Connection: keep-alive
Access-Control-Allow-Origin: *
Access-Control-Allow-Methods: GET, POST, DELETE, PUT
Access-Control-Allow-Headers: Content-Type, api_key, Authorization
Server: Jetty(9.2.9.v20150224)

{
  "id": 987,
  "category": {
    "id": 0,
    "name": "string"
  },
  "name": "duck",
  "photoUrls": [
    "string"
  ],
  "tags": [
    {
      "id": 0,
      "name": "string"
    }
  ],
  "status": "available"
}
987
PASSED: courseendproject1.Assignment001.assignment001Post

=====
Default test
Tests run: 1, Failures: 0, Skips: 0
=====

=====
Default suite
Total tests run: 1, Passes: 1, Failures: 0, Skips: 0
=====

```

```

package courseendproject1;

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 Assignment002 {

    Logger logger = LogManager.getLogger(Assignment002.class);

    @Test(priority='1')
    public void assignment002Post()
    {
        logger.info("Course End project - Assignment002 - POST request");
        File file = new File("resources/data.json");
        int id = RestAssured.given()
            .baseUrl("https://petstore.swagger.io/v2/pet")
            .contentType(ContentType.JSON)
            .body(file)
            .when().post()
            .then()
            .statusCode(200)
            .log().all()
            .body("name", Matchers.equalTo("duck")).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 = "assignment002Post")
    public void assignment002Put()
    {
        File file = new File("resources/dataput.json");
        int id = RestAssured.given()
            .baseUrl("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);

    }

}

```

<terminated> Assignment002 [TestNG] C:\Users\mnave\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.8.v20230

987

HTTP/1.1 200 OK

Date: Fri, 10 Nov 2023 11:38:40 GMT

Content-Type: application/json

Transfer-Encoding: chunked

Connection: keep-alive

Access-Control-Allow-Origin: *

Access-Control-Allow-Methods: GET, POST, DELETE, PUT

Access-Control-Allow-Headers: Content-Type, api_key, Authorization

Server: Jetty(9.2.9.v20150224)

```
{
  "id": 987,
  "category": {
    "id": 0,
    "name": "string"
  },
  "name": "duck",
  "photoUrls": [
    "string"
  ],
  "tags": [
    {
      "id": 0,
      "name": "string"
    }
  ],
  "status": "available_QA"
}
```

987

PASSED: courseendproject1.Assignment002.assignment002Put

PASSED: courseendproject1.Assignment002.assignment002Post

```
=====
Default test
Tests run: 2, Failures: 0, Skips: 0
=====
```

```
=====
Default suite
Total tests run: 2, Passes: 2, Failures: 0, Skips: 0
=====
```

◀

Assignment 003:

```
package courseendproject1;

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

import io.restassured.RestAssured;

public class Assignment003demo {

    @Test(priority='1')
    public void assignment003User()
    {
        RestAssured.given()
            .baseUrl("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));

    }

    @Test(priority='2')
    public void assignment004login()
    {
        RestAssured.given()
        .baseUrl("https://petstore.swagger.io/v2/user/login")
        .auth().preemptive().basic("Uname001", "@tt!tude")
        .when()
        .get()
        .then()
        .statusCode(200)
        .log().all()
        .body("message", Matchers.anything());

    }

}

```

```
Access-Control-Allow-Methods: GET, POST, DELETE, PUT
Access-Control-Allow-Headers: Content-Type, api_key, Authorization
Server: Jetty(9.2.9.v20150224)
```

```
{
  "id": 9223372036854756647,
  "username": "Username001",
  "firstName": "string",
  "lastName": "string",
  "email": "Positive@Attitude.com",
  "password": "string",
  "phone": "string",
  "userStatus": 1
}
```

```
HTTP/1.1 200 OK
Date: Fri, 10 Nov 2023 11:39:52 GMT
Content-Type: application/json
Transfer-Encoding: chunked
Connection: keep-alive
Access-Control-Allow-Origin: *
Access-Control-Allow-Methods: GET, POST, DELETE, PUT
Access-Control-Allow-Headers: Content-Type, api_key, Authorization
Expires-After: Fri Nov 10 12:39:52 UTC 2023
Rate-Limit: 5000
Server: Jetty(9.2.9.v20150224)
```

```
{
  "code": 200,
  "type": "unknown",
  "message": "logged in user session:1699616392840"
}
```

```
SSHD: courseendproject1.Assignment003demo.assignment003User
SSHD: courseendproject1.Assignment003demo.assignment004login
```

```
=====
Default test
Tests run: 2, Failures: 0, Skips: 0
=====
```

```
=====
Default suite
Total tests run: 2, Passes: 2, Failures: 0, Skips: 0
=====
```

```
package courseendproject1;
```

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

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

```
public class Assignment005and006 {
```

```
    @Test(priority='1')
    public void assignment005FindByStatus()
    {
        RestAssured.given()
            .baseUrl("https://petstore.swagger.io/v2/pet/findByStatus")
            //.queryParams("status", "available")
            //.queryParams("status", "pending")
            .queryParams("status", "sold")
            .when()
            .get()
            .then()
            .statusCode(200)
            .log().all()

        ;
    }
}
```

```
}
```

```
@Test(priority='2')  
public void assignment006Logout()  
{  
    RestAssured.given()  
        .baseUrl("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"))  
    ;  
}
```

```
}
```

```
}
```

terminated> Assignment005and006 [testNG] C:\Users\mnave\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.v

```
[
  {
    "id": 9223372016900014000,
    "category": {
      "id": 0,
      "name": "string"
    },
    "name": "fish",
    "photoUrls": [
      "string"
    ],
    "tags": [
      {
        "id": 0,
        "name": "string"
      }
    ],
    "status": "sold"
  }
]
HTTP/1.1 200 OK
Date: Fri, 10 Nov 2023 11:40:54 GMT
Content-Type: application/json
Transfer-Encoding: chunked
Connection: keep-alive
Access-Control-Allow-Origin: *
Access-Control-Allow-Methods: GET, POST, DELETE, PUT
Access-Control-Allow-Headers: Content-Type, api_key, Authorization
Server: Jetty(9.2.9.v20150224)

{
  "code": 200,
  "type": "unknown",
  "message": "ok"
}
PASSED: courseendproject1.Assignment005and006.assignment006Logout
PASSED: courseendproject1.Assignment005and006.assignment005FindByStatus
```

```
=====
Default test
Tests run: 2, Failures: 0, Skips: 0
=====
```

```
package courseendproject1;
```

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

```
public class Log4jDemo {
```

```
    Logger logger = LogManager.getLogger(Log4jDemo.class);
```

```
    @Test
    public void method1()
    {
        System.out.println("this is method1");
    }
}
```



```

        logger.trace(" This is trace method");

        logger.info(" this is information message");

        logger.error(" this is an error message");

        logger.fatal(" this is a fatal messgae");

        logger.warn(" this is a warning message");

        System.out.println("this is end of method1");
    }
}

```

The screenshot shows an IDE with a Java file named `Log4jDemo.java` and its execution output. The code defines a `Log4jDemo` class with a `method1()` that uses `LogManager` to create a `logger` and prints various log messages. The output window shows the execution of `method1()` with the following log messages:

```

<terminated> Log4jDemo [TestNG] C:\Users\mnave\p2\pool\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64_17.0.8.v20231110\jre\bin\java.exe
SLF4J: Defaulting to no-operation (NOP) logger implementation
SLF4J: See http://www.slf4j.org/codes.html#StaticLoggerBinder for further details.
this is method1
[INFO ] 2023-11-10 17:12:38.052 [main] Log4jDemo - this is information message
[ERROR] 2023-11-10 17:12:38.056 [main] Log4jDemo - this is an error message
[FATAL] 2023-11-10 17:12:38.056 [main] Log4jDemo - this is a fatal messgae
[WARN ] 2023-11-10 17:12:38.056 [main] Log4jDemo - this is a warning message
this is end of method1
PASSED: courseendproject1.Log4jDemo.method1

=====
Default test
Tests run: 1, Failures: 0, Skips: 0
=====

Default suite
Total tests run: 1, Passes: 1, Failures: 0, Skips: 0
=====

```

Pom File for above source code:

```

<?xml version="1.0" encoding="UTF-8"?>

<project xmlns="http://maven.apache.org/POM/4.0.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
    http://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>

  <groupId>ATE-phase3-RestAssured</groupId>
  <artifactId>ATE-phase3-RestAssured</artifactId>
  <version>0.0.1-SNAPSHOT</version>

  <name>ATE-phase3-RestAssured</name>
  <!-- FIXME change it to the project's website -->
  <url>http://www.example.com</url>

  <properties>
    <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
    <maven.compiler.source>1.7</maven.compiler.source>
    <maven.compiler.target>1.7</maven.compiler.target>
  </properties>

```

```
<dependencies>

    <dependency>
        <groupId>junit</groupId>
        <artifactId>junit</artifactId>
        <version>4.11</version>
        <scope>test</scope>
    </dependency>

    <!-- https://mvnrepository.com/artifact/io.cucumber/cucumber-java -->
<dependency>
    <groupId>io.cucumber</groupId>
    <artifactId>cucumber-java</artifactId>
    <version>7.10.1</version>
</dependency>

    <!-- https://mvnrepository.com/artifact/io.cucumber/cucumber-java -->
<dependency>
    <groupId>io.cucumber</groupId>
    <artifactId>cucumber-junit</artifactId>
    <version>7.10.1</version>
    <scope>test</scope>
</dependency>

    <!-- https://mvnrepository.com/artifact/io.cucumber/cucumber-java -->
<dependency>
    <groupId>io.cucumber</groupId>
    <artifactId>cucumber-core</artifactId>
    <version>7.10.1</version>
</dependency>

    <!-- https://mvnrepository.com/artifact/io.rest-assured/rest-assured -->
<dependency>
    <groupId>io.rest-assured</groupId>
    <artifactId>rest-assured</artifactId>
    <version>5.3.0</version>
    <scope>test</scope>
</dependency>

    <!-- https://mvnrepository.com/artifact/io.rest-assured/rest-assured-common -->
<dependency>
    <groupId>io.rest-assured</groupId>
    <artifactId>rest-assured-common</artifactId>
    <version>5.3.0</version>
</dependency>

    <!-- https://mvnrepository.com/artifact/io.rest-assured/rest-assured-all -->
<dependency>
    <groupId>io.rest-assured</groupId>
    <artifactId>rest-assured-all</artifactId>
    <version>5.3.0</version>
    <scope>test</scope>
</dependency>

    <!-- https://mvnrepository.com/artifact/io.rest-assured/json-path -->
<dependency>
    <groupId>io.rest-assured</groupId>
    <artifactId>json-path</artifactId>
    <version>5.3.0</version>
    <scope>test</scope>
</dependency>

    <!-- https://mvnrepository.com/artifact/org.json/json -->
```

```

<dependency>
  <groupId>org.json</groupId>
  <artifactId>json</artifactId>
  <version>20180813</version>
</dependency>

<!-- https://mvnrepository.com/artifact/org.hamcrest/hamcrest -->
<dependency>
  <groupId>org.hamcrest</groupId>
  <artifactId>hamcrest</artifactId>
  <version>2.2</version>
  <scope>test</scope>
</dependency>

<!-- https://mvnrepository.com/artifact/com.fasterxml.jackson.core/jackson-databind -->
<dependency>
  <groupId>com.fasterxml.jackson.core</groupId>
  <artifactId>jackson-databind</artifactId>
  <version>2.15.2</version>
</dependency>

  <!-- https://mvnrepository.com/artifact/org.testng/testng -->
<dependency>
  <groupId>org.testng</groupId>
  <artifactId>testng</artifactId>
  <version>7.7.1</version>
  <scope>test</scope>
</dependency>

<dependency>
  <groupId>org.apache.logging.log4j</groupId>
  <artifactId>log4j-api</artifactId>
  <version>2.14.1</version>
</dependency>

<!-- https://mvnrepository.com/artifact/org.apache.logging.log4j/log4j-core -->
<dependency>
  <groupId>org.apache.logging.log4j</groupId>
  <artifactId>log4j-core</artifactId>
  <version>2.14.1</version>
</dependency>

</dependencies>

<build>
  <pluginManagement><!-- lock down plugins versions to avoid using Maven defaults
(may be moved to parent pom) -->
    <plugins>
      <!-- clean lifecycle, see https://maven.apache.org/ref/current/maven-core/lifecycles.html#clean_Lifecycle -->
      <plugin>
        <artifactId>maven-clean-plugin</artifactId>
        <version>3.1.0</version>
      </plugin>
      <!-- default lifecycle, jar packaging: see
https://maven.apache.org/ref/current/maven-core/default-bindings.html#Plugin_bindings_for_jar_packaging -->
      <plugin>
        <artifactId>maven-resources-plugin</artifactId>
        <version>3.0.2</version>
      </plugin>
      <plugin>
        <artifactId>maven-compiler-plugin</artifactId>
        <version>3.8.0</version>
      </plugin>
    </plugins>
  </pluginManagement>
</build>

```

```

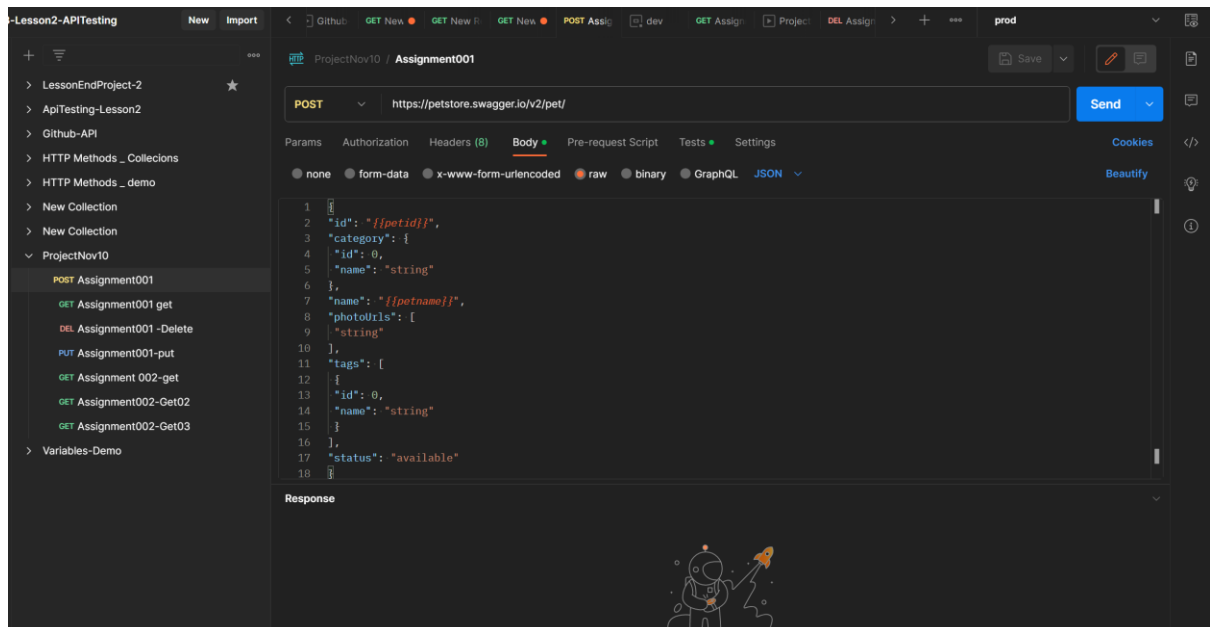
        <artifactId>maven-surefire-plugin</artifactId>
        <version>2.22.1</version>
    </plugin>
    <plugin>
        <artifactId>maven-jar-plugin</artifactId>
        <version>3.0.2</version>
    </plugin>
    <plugin>
        <artifactId>maven-install-plugin</artifactId>
        <version>2.5.2</version>
    </plugin>
    <plugin>
        <artifactId>maven-deploy-plugin</artifactId>
        <version>2.8.2</version>
    </plugin>
    <!-- site lifecycle, see https://maven.apache.org/ref/current/maven-
core/lifecycles.html#site_Lifecycle -->
    <plugin>
        <artifactId>maven-site-plugin</artifactId>
        <version>3.7.1</version>
    </plugin>
    <plugin>
        <artifactId>maven-project-info-reports-plugin</artifactId>
        <version>3.0.0</version>
    </plugin>
</plugins>
</pluginManagement>
</build>
</project>

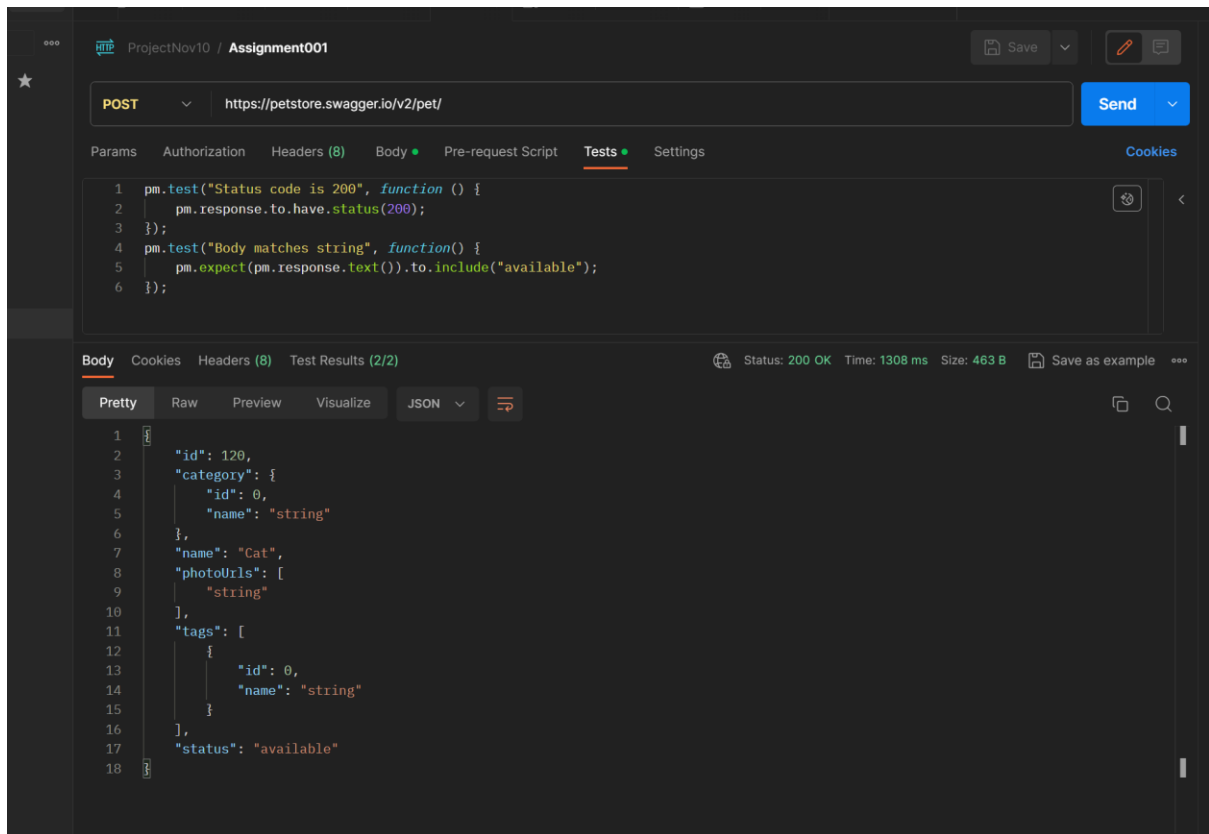
```

POSTMAN

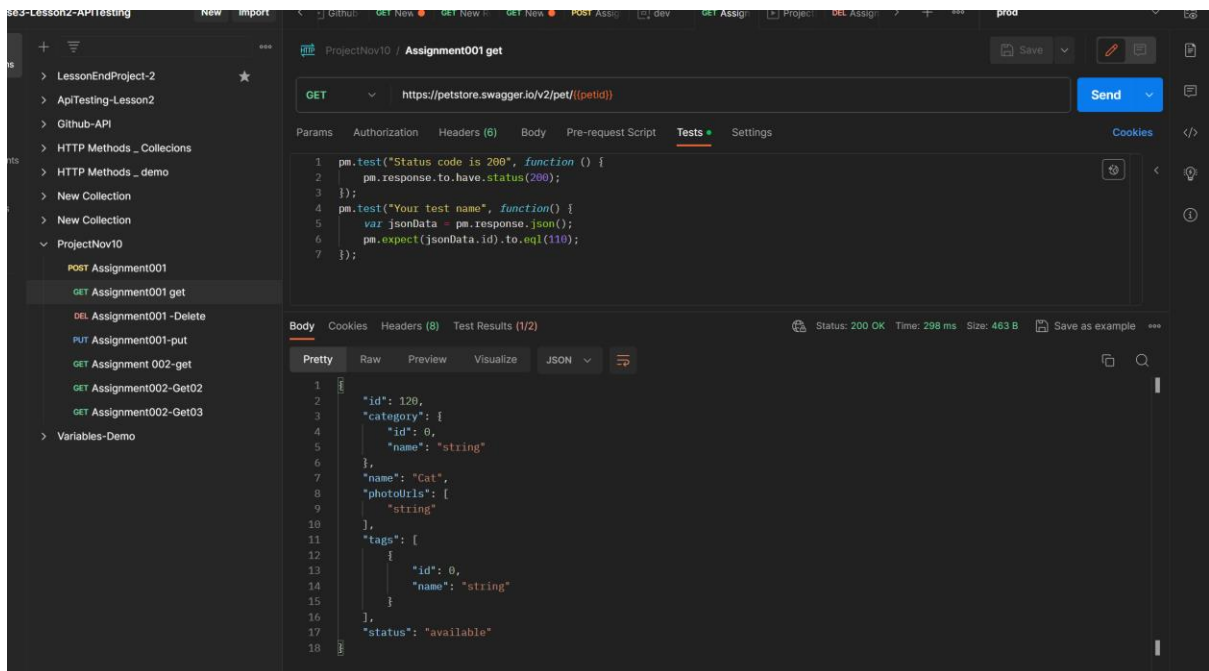
Created Collections and given method POST , Url to update the data.

Creating petid and petname.





Postman - Asssignment 002



Postman-Assignment Delete Method

DELETE

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

Params

Authorization

Headers (6)

Body

Pre-request Script

Tests

Settings

```
1 pm.test("Status code is 200", function () {
2   pm.response.to.have.status(200);
3 });
```

Body

Cookies

Headers (8)

Test Results (1/1)

Status: 200 OK

Time:

Pretty

Raw

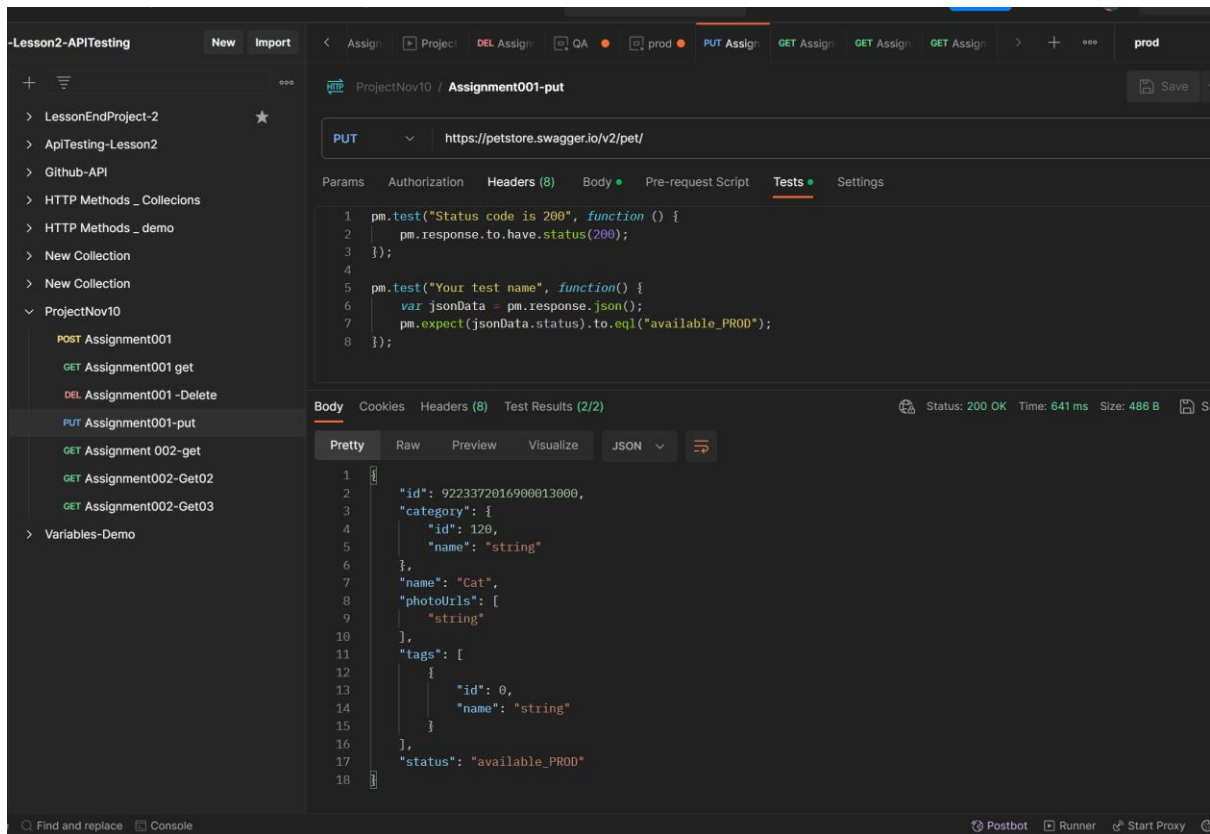
Preview

Visualize

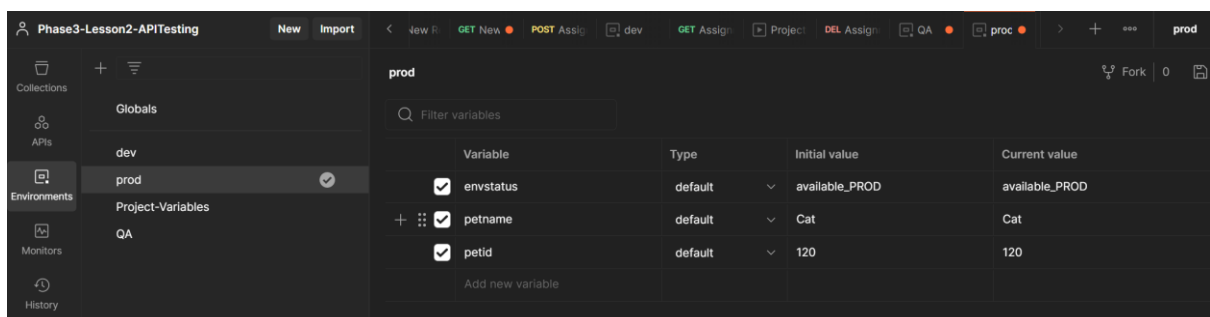
JSON

```
1 {
2   "code": 200,
3   "type": "unknown",
4   "message": "120"
5 }
```

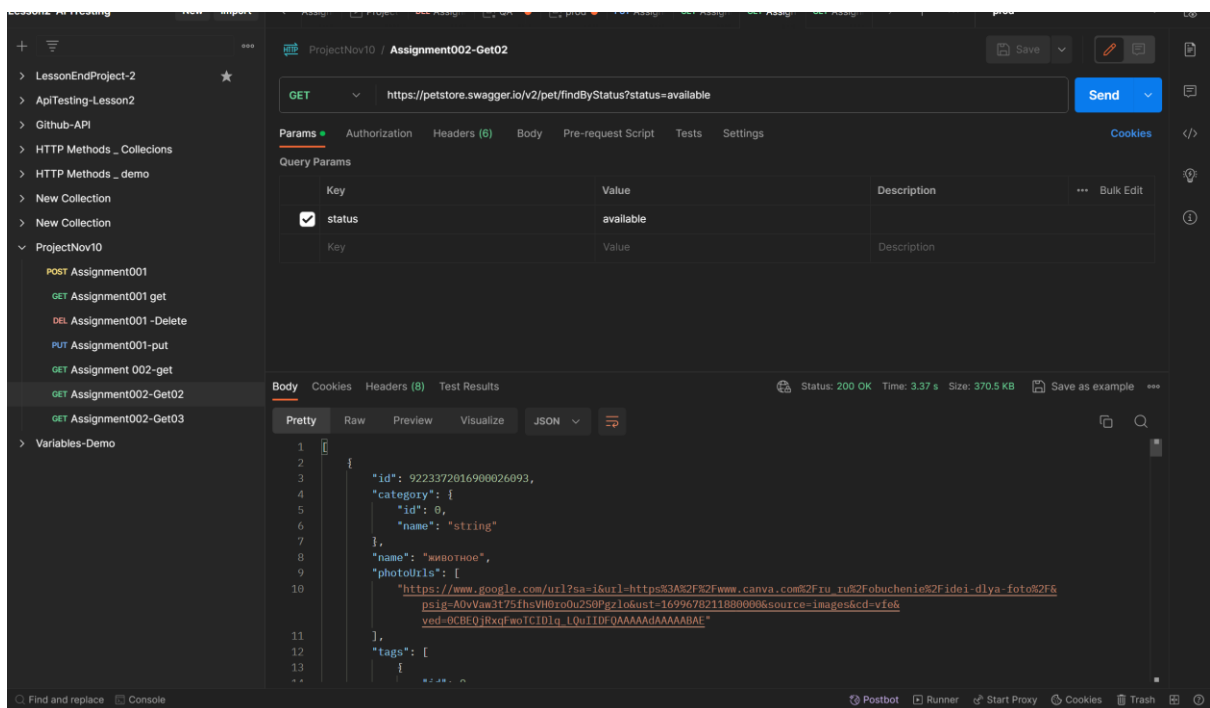
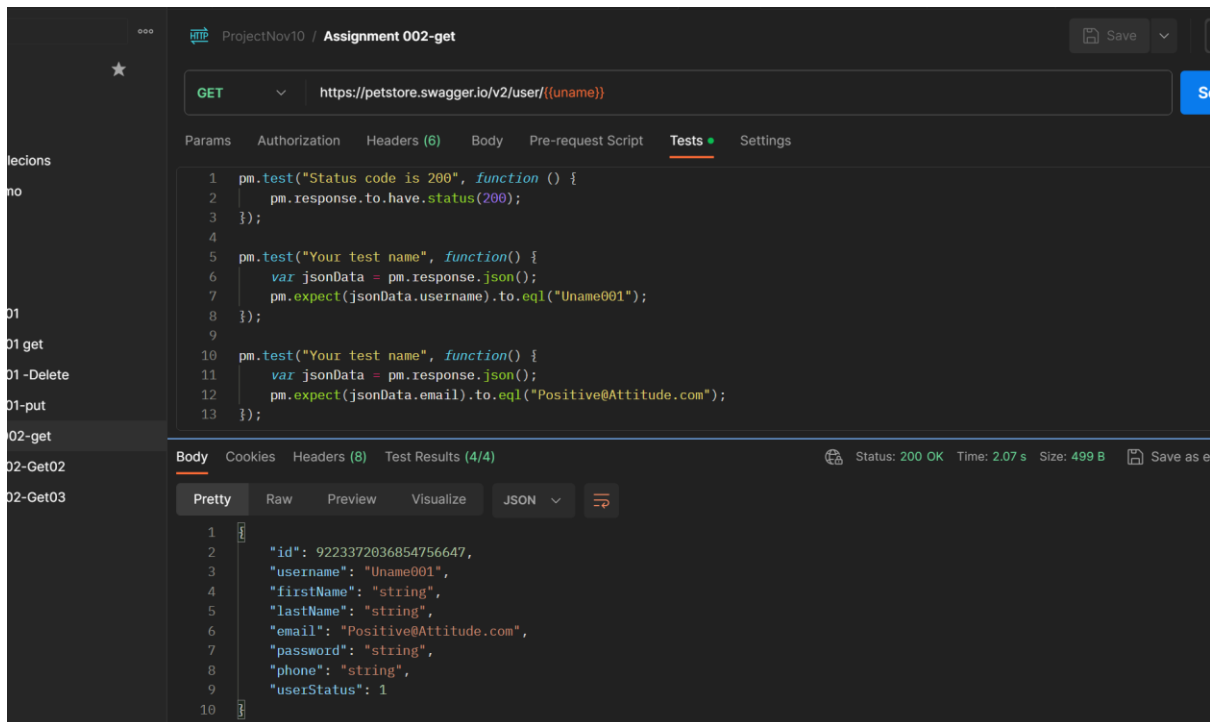
Put Method-Asssignment001



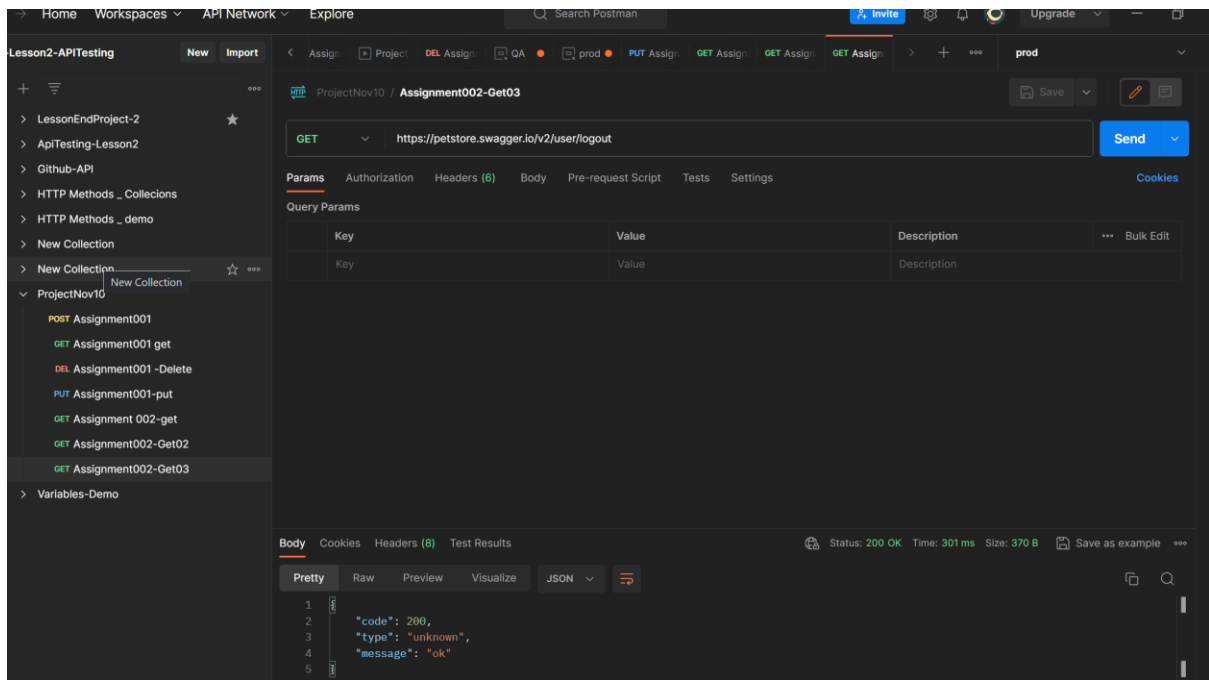
Environment Variables..of Methods to update data.



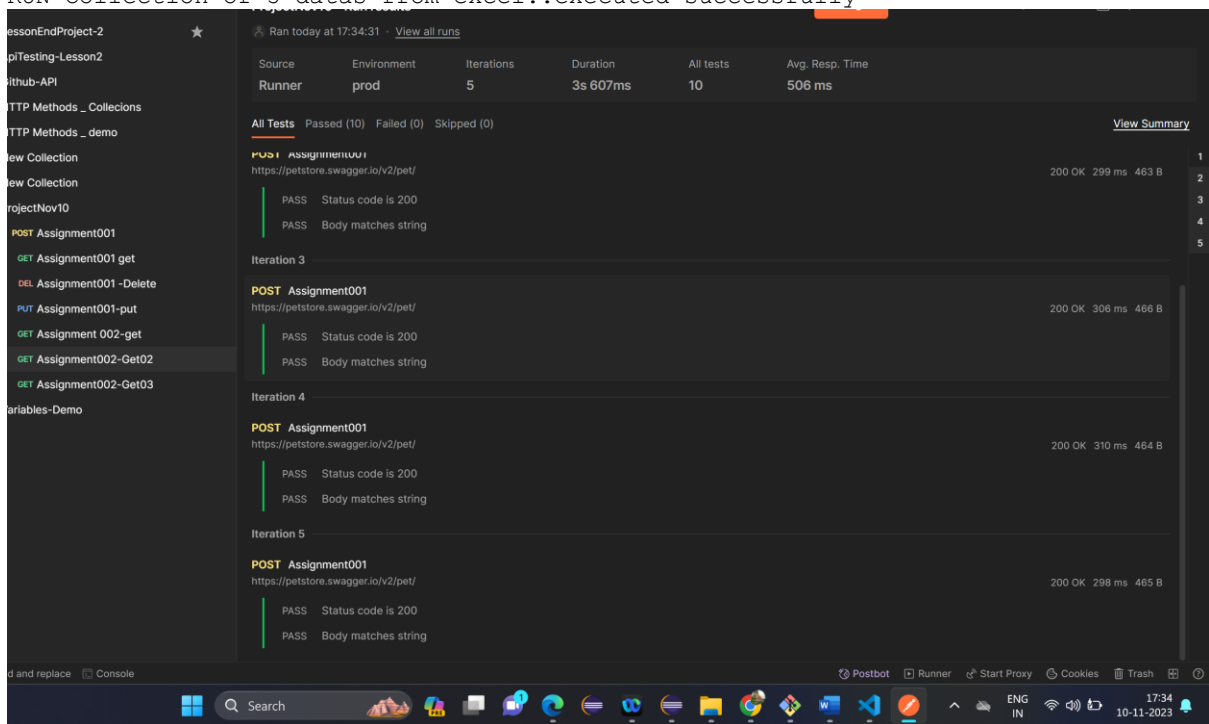
Creating Username And updating



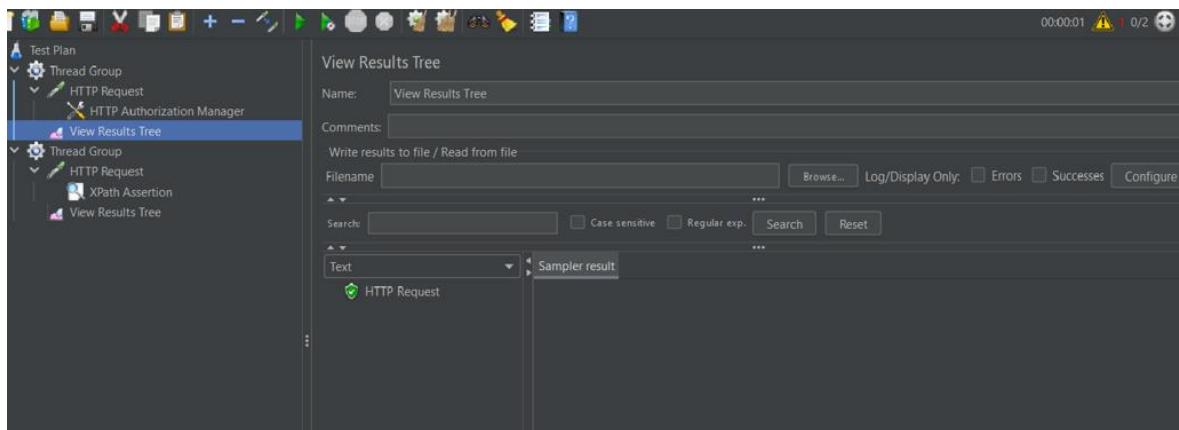
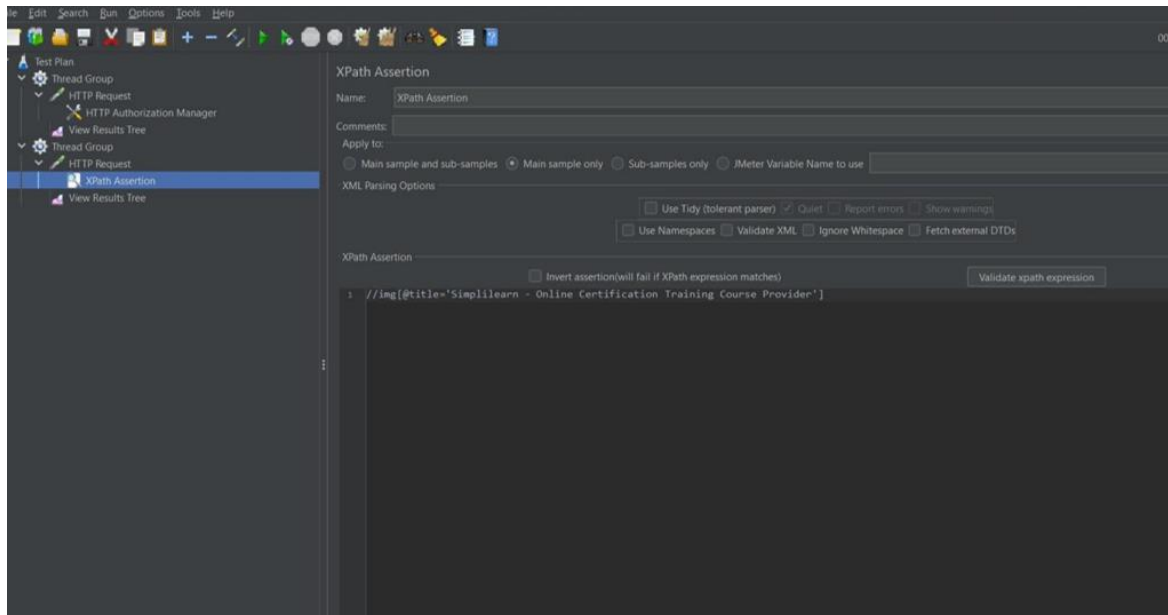
Logout of url server page



RUN Collection of 5 datas from excel..executed successfully



JMETER



Test Plan

- Thread Group
 - HTTP Request
 - HTTP Authorization Manager
 - View Results Tree
- Thread Group
 - HTTP Request
 - XPath Assertion
 - View Results Tree

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.

Text

HTTP Request

Sampler result Request Response data

Thread Name: Thread Group 1-1
Sample Start: 2023-11-09 13:37:00 IST
Load time: 1675
Connect Time: 956
Latency: 1673
Size in bytes: 276
Sent bytes: 176
Headers size in bytes: 229
Body size in bytes: 47
Sample Count: 1
Error Count: 0
Data type ("text"/"bin"): text
Response code: 200
Response message: OK

HTTPSampleResult fields:
ContentType: application/json
DataEncoding: null

Test Plan

- Thread Group
 - HTTP Request
 - HTTP Authorization Manager
 - View Results Tree
- Thread Group
 - HTTP Request
 - XPath Assertion
 - 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
https://httpbin.org	user	*****	

Test Plan

Thread Group

HTTP Request

HTTP Authorization Manager

View Results Tree

Thread Group

HTTP Request

XPath Assertion

View Results Tree

HTTP Request

Name: HTTP Request

Comments:

BasicAdvanced

Web Server

Protocol (http):httpsServer Name or IP:www.simplilearn.comPort Number:

HTTP Request

GETPath: /Content encoding:

☐ Redirect Automatically

☒ Follow Redirects

☒ Use KeepAlive

☐ Use multipart/form-data

☐ Browser-compatible headers

ParametersBody DataFiles Upload

Send Parameters With the Request:

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