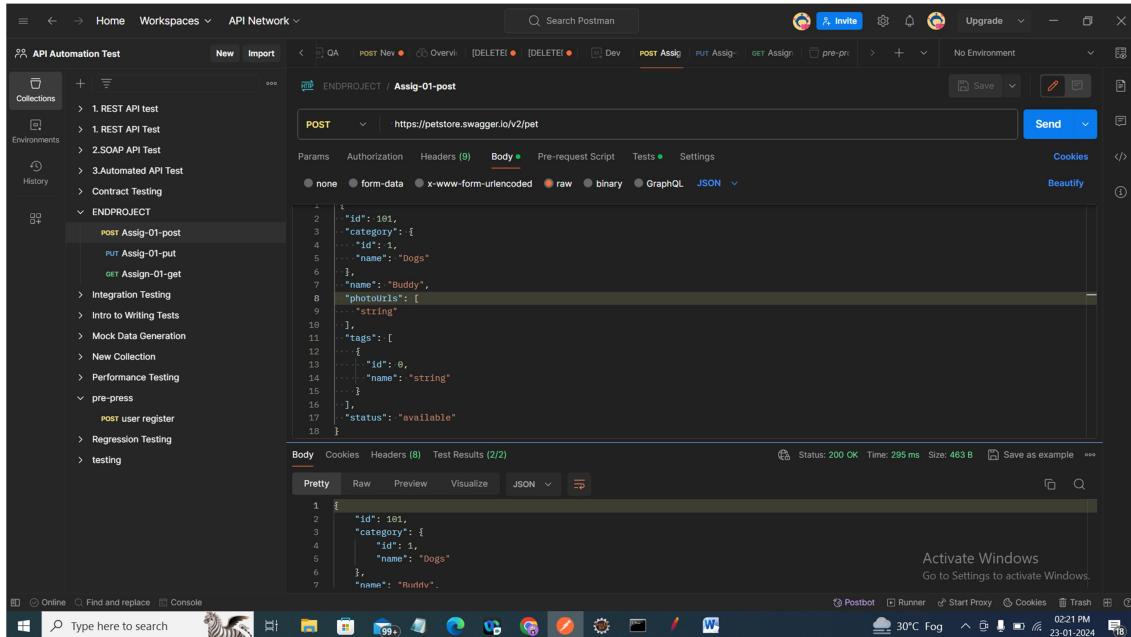


Screenshots

Postman Framework Test



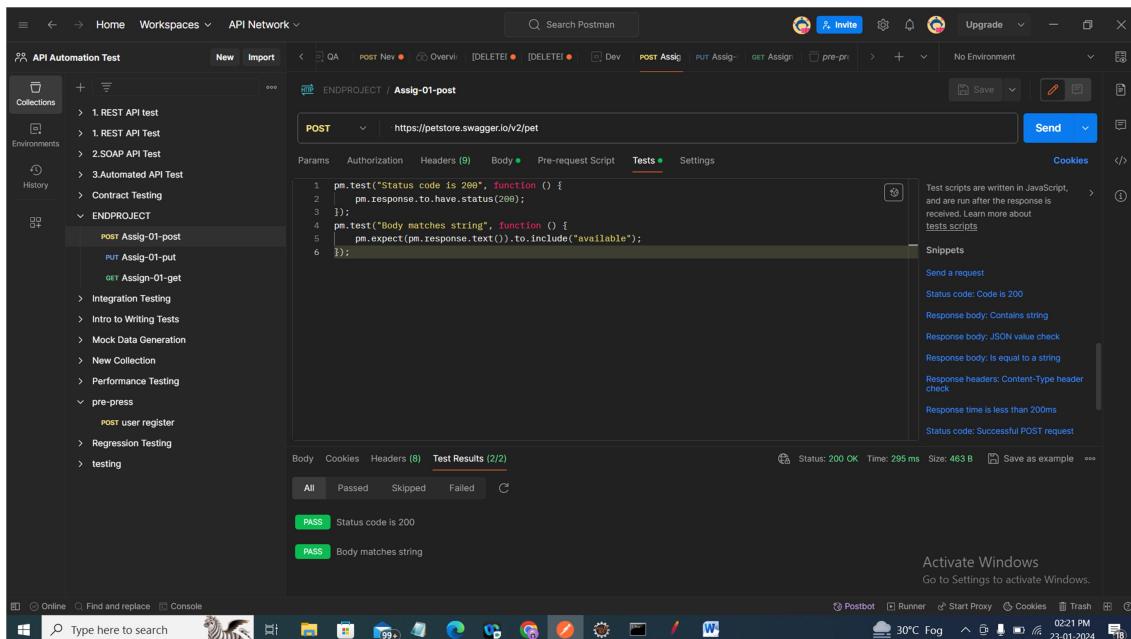
The screenshot shows the Postman interface with a dark theme. On the left, the sidebar displays a collection named "ENDPROJECT / Assig-01-post" containing several API endpoints: "POST Assig-01-post", "PUT Assig-01-put", and "GET Assig-01-get". The "POST Assig-01-post" endpoint is selected. The main panel shows a POST request to "https://petstore.swagger.io/v2/pet". The "Body" tab is active, showing a JSON payload:

```
1 {
2     "id": 101,
3     "category": {
4         "id": 1,
5         "name": "Dogs"
6     },
7     "name": "Buddy",
8     "photoUrls": [
9         ...
10        "string"
11    ],
12    "tags": [
13        ...
14        {
15            "id": 0,
16            "name": "string"
17        }
18    ],
19    "status": "available"
}
```

The response section shows a status of 200 OK with a response time of 295 ms and a size of 463 B. Below the response, the "Tests" tab is open, displaying a snippet of JavaScript code:

```
1 pm.test("Status code is 200", function () {
2     pm.response.to.have.status(200);
3 });
4 pm.test("Body matches string", function () {
5     pm.expect(pm.response.text()).to.include("available");
6 });
```

The status bar at the bottom indicates "Activate Windows" and shows the date and time as 23-01-2024.



This screenshot is identical to the one above, showing the same API call and response. The difference is in the "Tests" tab, which now displays the results of the executed tests:

- Test 1: Status code is 200 - Status: PASS
- Test 2: Body matches string - Status: PASS

The status bar at the bottom indicates "Activate Windows" and shows the date and time as 23-01-2024.

The screenshot shows the Postman application interface. On the left, there's a sidebar with a tree view of API tests, environments, and history. The main area displays a test collection named "ENDPROJECT / Assig-01-put". A specific PUT request to "https://petstore.swagger.io/v2/pet" is selected. The "Tests" tab is active, showing a JavaScript code block that performs assertions on the response body. The "Body" tab shows the JSON payload sent to the server. The "Test Results" section at the bottom indicates a successful run with three green "PASS" status indicators. The status bar at the bottom right shows "Status: 200 OK Time: 250 ms Size: 458 B".

This screenshot shows the same Postman interface as the first one, but with a different JSON payload in the "Body" tab. The payload is a complex object with nested properties such as "category" and "tags". The "Pretty" tab is selected in the "Body" tab, which displays the JSON in a readable, indented format. The "Test Results" section shows a single green "PASS" status indicator. The status bar at the bottom right shows "Status: 200 OK Time: 250 ms Size: 458 B".

The screenshot shows the Postman application interface. On the left, there's a sidebar with 'Collections' (API Automation Test), 'Environments', and 'History'. The main area displays a 'GET Assign-01-get' request. The URL is <https://petstore.swagger.io/v2/pet/findByStatus?status=available>. The 'Body' tab shows a JSON payload:

```
1 {
  "id": 0,
  "category": {
    "id": 0,
    "name": "string"
  },
  "name": "doggie",
  "photoUrls": [
    "string"
  ],
  "tags": [
    {
      "id": 0,
      "name": "string"
    }
  ],
  "status": "available"
}
```

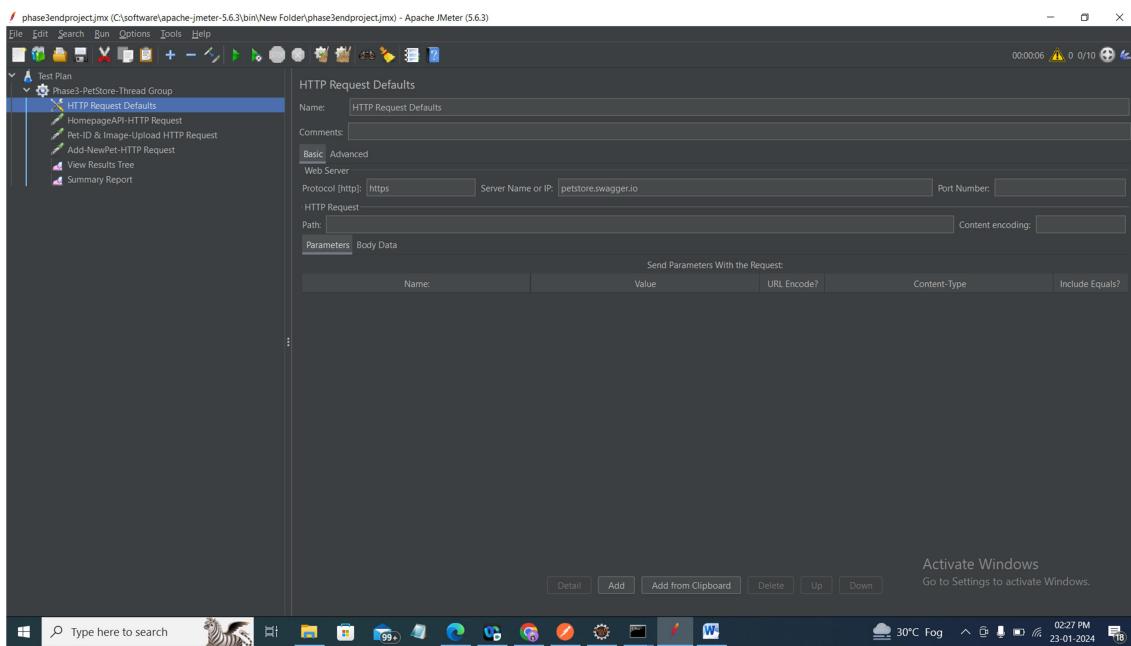
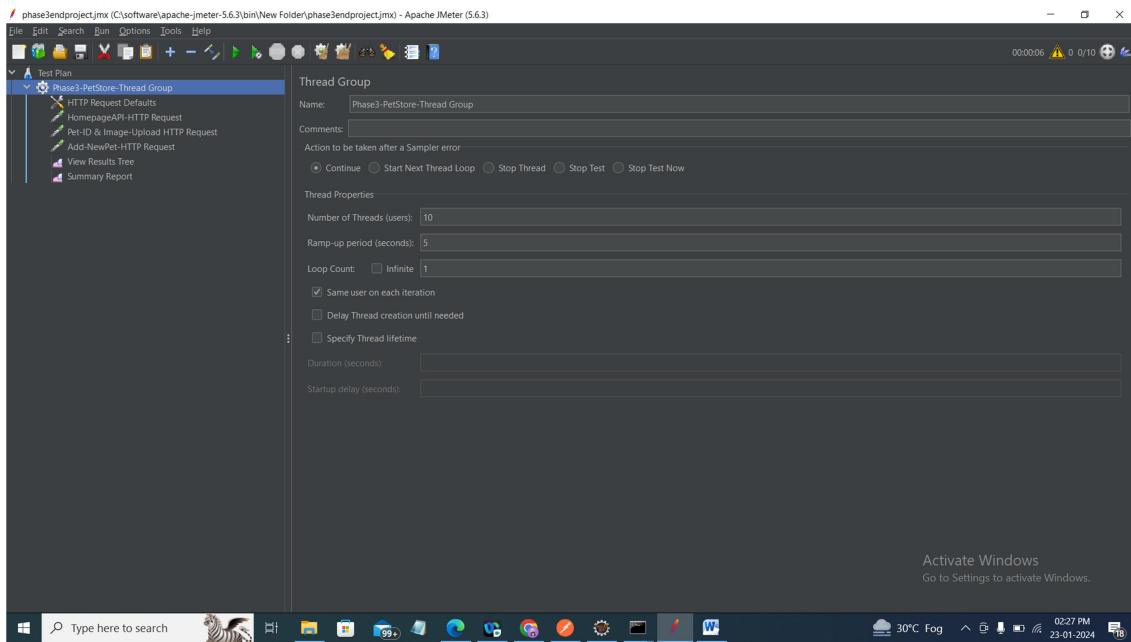
The response status is 200 OK, Time: 1202 ms, Size: 35.15 KB. Below the response, the 'Test Results' section shows two green 'PASS' status messages: 'Status code is 200' and 'Body matches string'.

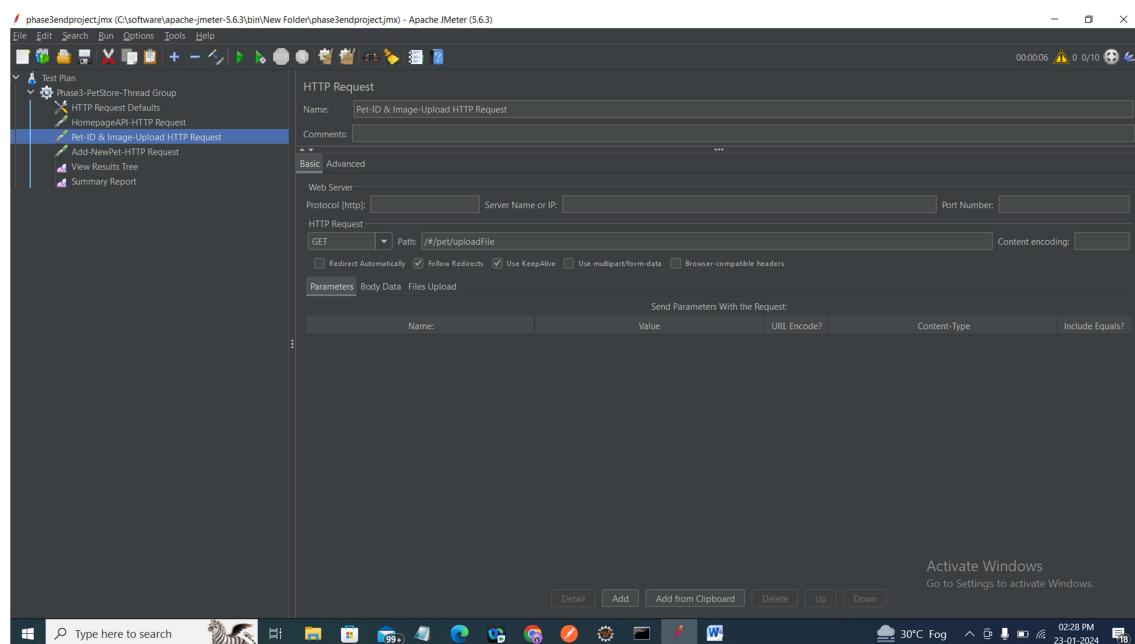
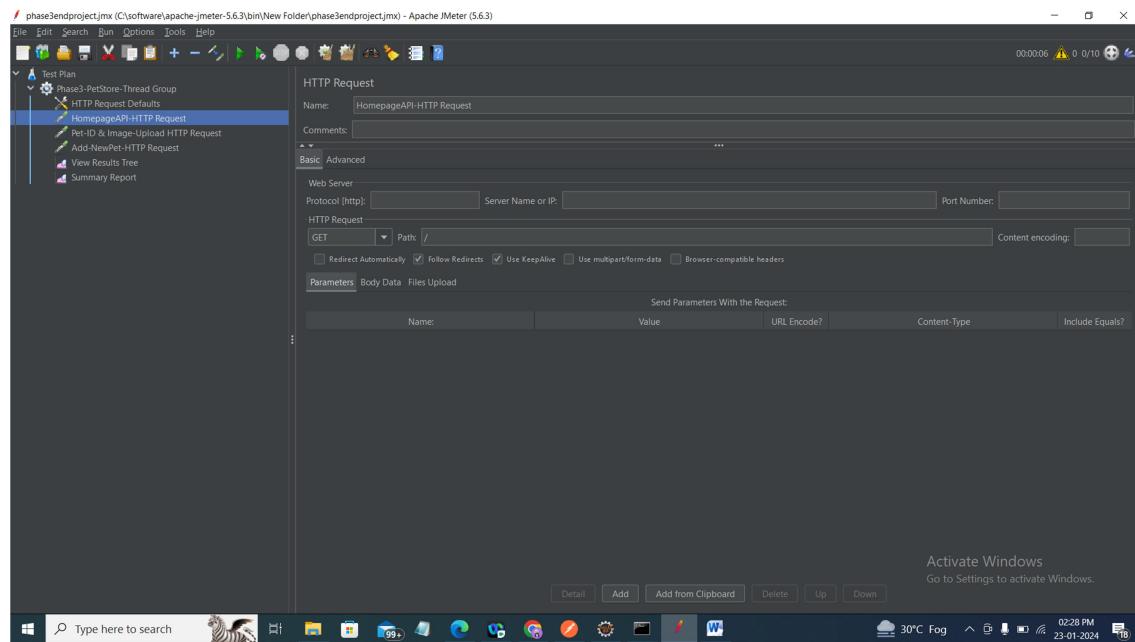
This screenshot is identical to the one above, but the 'Tests' tab is selected in the top navigation bar. A test script is present in the 'Tests' section:

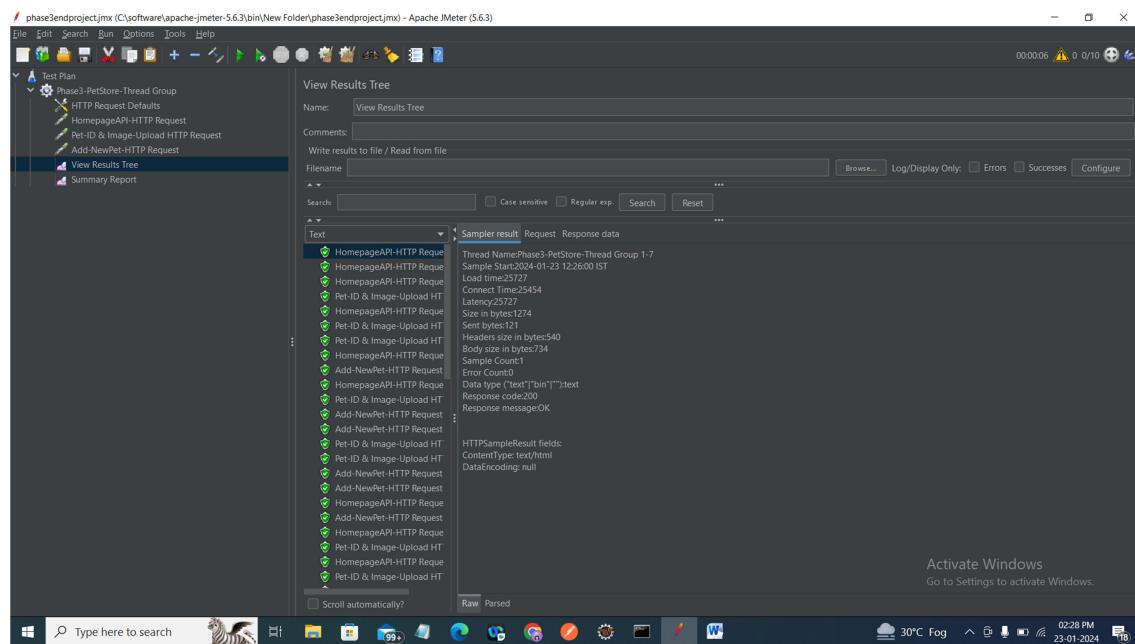
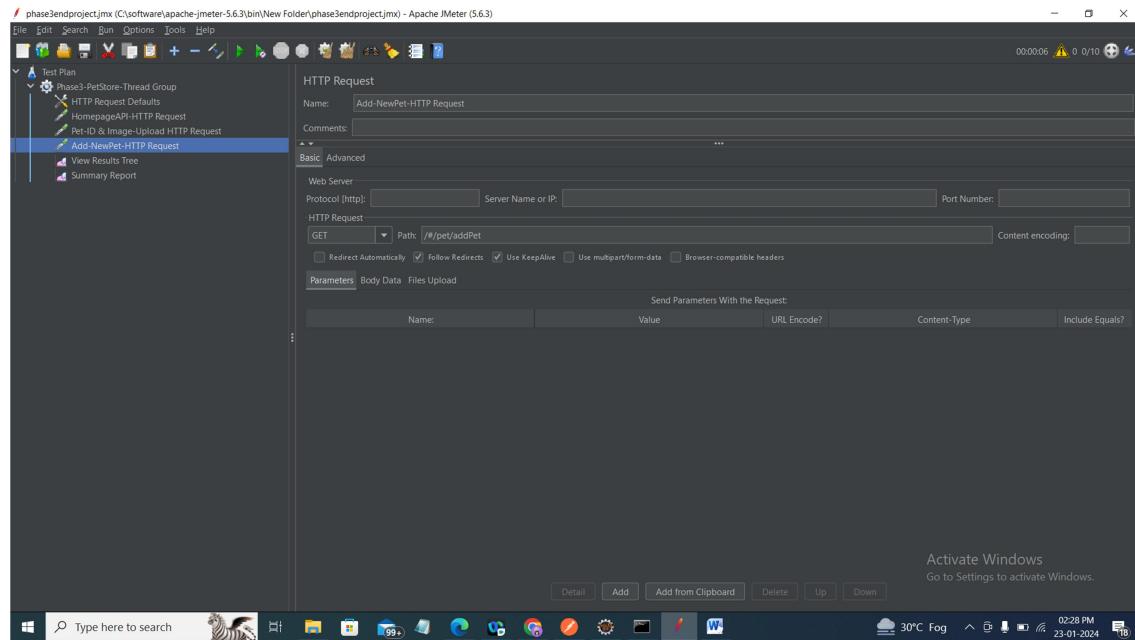
```
1 pm.test("Status code is 200", function () {
2   pm.response.to.have.status(200);
3 });
4 pm.test("Body matches string", function () {
5   pm.expect(pm.response.text()).to.include("available");
6 });
```

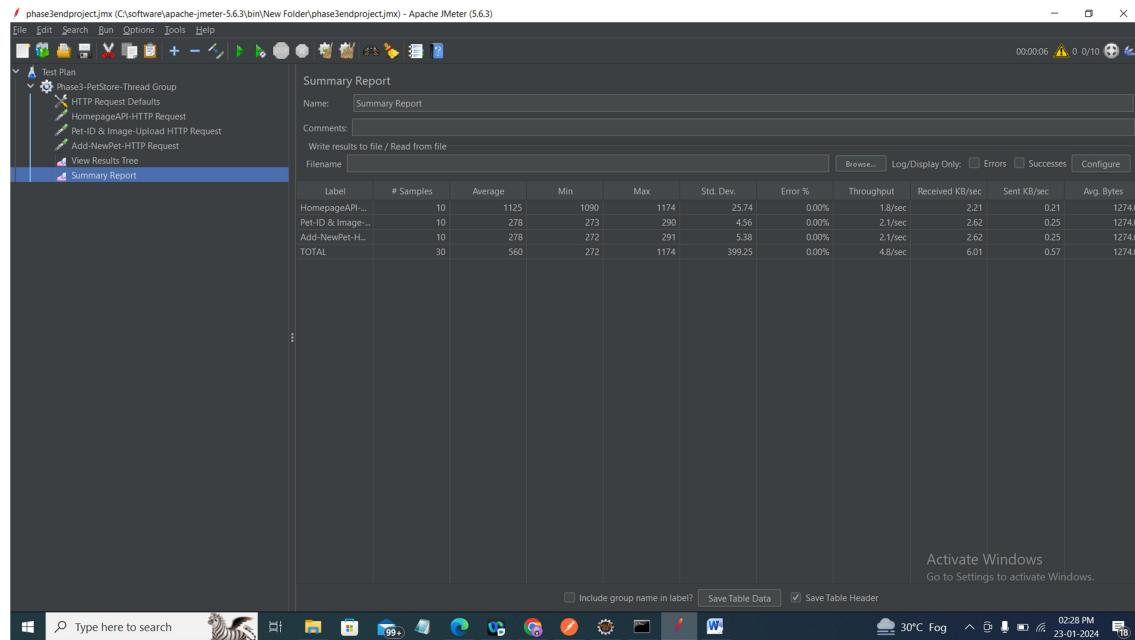
The right side of the interface shows a sidebar with various test snippets and environment-related options.

Jmeter Framework Test









Rest-Assured Framework Test

```

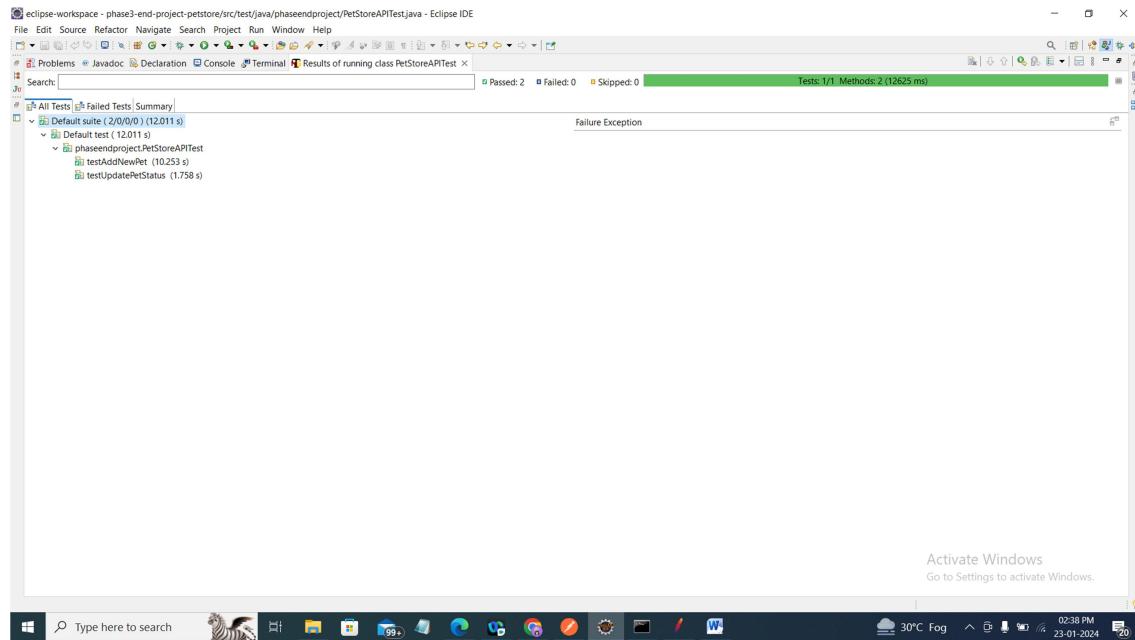
package phase3endproject;
import org.testng.annotations.Test;

public class PetStoreAPITest {
    @Test
    public void testGetPetById() {
        RestAssured.baseURI = "https://petstore.swagger.io/v2";
        given()
            .pathParam("petId", 0)
        .when()
            .get("/pet/{petId}")
        .then()
            .statusCode(200)
            .body("id", equalTo(0))
            .body("name", equalTo("Fido"))
            .body("status", equalTo("available"));
    }

    @Test
    public void testAddNewPet() {
        RestAssured.baseURI = "https://petstore.swagger.io/v2";
        given()
            .header("Content-Type", "application/json")
            .body("{ \"id\": 101, \"category\": { \"id\": 1, \"name\": \"Dogs\" }, \"name\": \"Buddy\", \"photoUrl\": \"\", \"status\": \"available\" }")
        .when()
            .post("/pet")
    }
}

```

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



```
C:\Windows\system32\cmd.exe
ENDPROJECT
+ Assig-01-post
POST https://petstore.swagger.io/v2/pet [200 OK, 4618, 1434ms]
✓ Status code is 200
✓ Body matches string

+ Assig-01-put
PUT https://petstore.swagger.io/v2/pet [200 OK, 458B, 292ms]
✓ Status code is 200
✓ Body matches string
✓ Your test name

+ Assig-01-get
GET https://petstore.swagger.io/v2/pet/findByStatus?status=available [200 OK, 43.33kB, 553ms]
✓ Status code is 200
✓ Body matches string



|                    | executed | failed |
|--------------------|----------|--------|
| iterations         | 1        | 0      |
| requests           | 3        | 0      |
| test-scripts       | 3        | 0      |
| prerequest-scripts | 0        | 0      |
| assertions         | 7        | 0      |


total run duration: 2.8s
total data received: 43.28kB (approx)
average response time: 759ms [min: 292ms, max: 1434ms, s.d.: 408ms]

C:\Users\PC\OneDrive\Documents\WORK\Postman

Activate Windows
Go to Settings to activate Windows.
```

The screenshot shows a command prompt window titled "cmd.exe" with the path "C:\Windows\system32". It displays the execution of three API tests using the Postman collection named "Assig-01". The first test, "Assig-01-post", is a POST request to "https://petstore.swagger.io/v2/pet" and passes with a 200 OK status code. The second test, "Assig-01-put", is a PUT request to the same endpoint and also passes with a 200 OK status code. The third test, "Assig-01-get", is a GET request to "https://petstore.swagger.io/v2/pet/findByStatus?status=available" and passes with a 200 OK status code. Below the tests, a summary table provides statistics: 1 iteration, 3 requests, 3 test-scripts, 0 prerequest-scripts, and 7 assertions. The total run duration is 2.8 seconds. The taskbar at the bottom shows various icons and the system tray with the Windows logo, a search bar, and the date/time "23-01-2024 02:40 PM".