# Task :

**GitHub API Assignment (L2 candidate)**

* Instructions:
  + Review the documentation of the GitHub API at <<https://developer.github.com/v3/>>. Imagine that you are on the team that has been tasked with implementing this specification.
  + Part 1: Answer the following questions:
    - What concerns would you have from a testing perspective?
    - How would you go about tackling the QA for this work?
    - What sort of tests would be worth describing or worth automating?
    - What tools would you use?
    - See Part 2 for how you will provide us with your answers to these questions.
  + Part 2: Stub implementation
    - Create a test suite for at least one endpoint with the tech stack of your choice, and provide the URL of a public git repository that contains your implementation and tests.
    - Include documentation in your repository that contains your written answers to the questions above.

# Solution:

## Strategy:

1. As a Scrum team, we will need to agree on a testing strategy. Our task – is a functional testing, we need to make sure that the API is functioning as expected.
2. We need to make sure, that the API implemented and works accordingly to the requirement documentation.

## Testing Stages:

1. Need to verify HTTP status codes.
2. Verify response payload.
3. Verify response headers.

## Test scenario categories:

1. Positive tests.
2. Extended positive testing (with additional parameters)
3. Negative testing with valid parameters
4. Negative testing with invalid parameters

## Test scenarios worth automating:

1. The most used interfaces
2. Find the API that will do the most damage to the app if this API interface is down.

## Tech stack for the API testing:

1. Postman
2. Java
3. Maven
4. Cucumber Open with JUnit
5. GIT with GitHub
6. Chrome Browser
7. IntelliJ IDEA

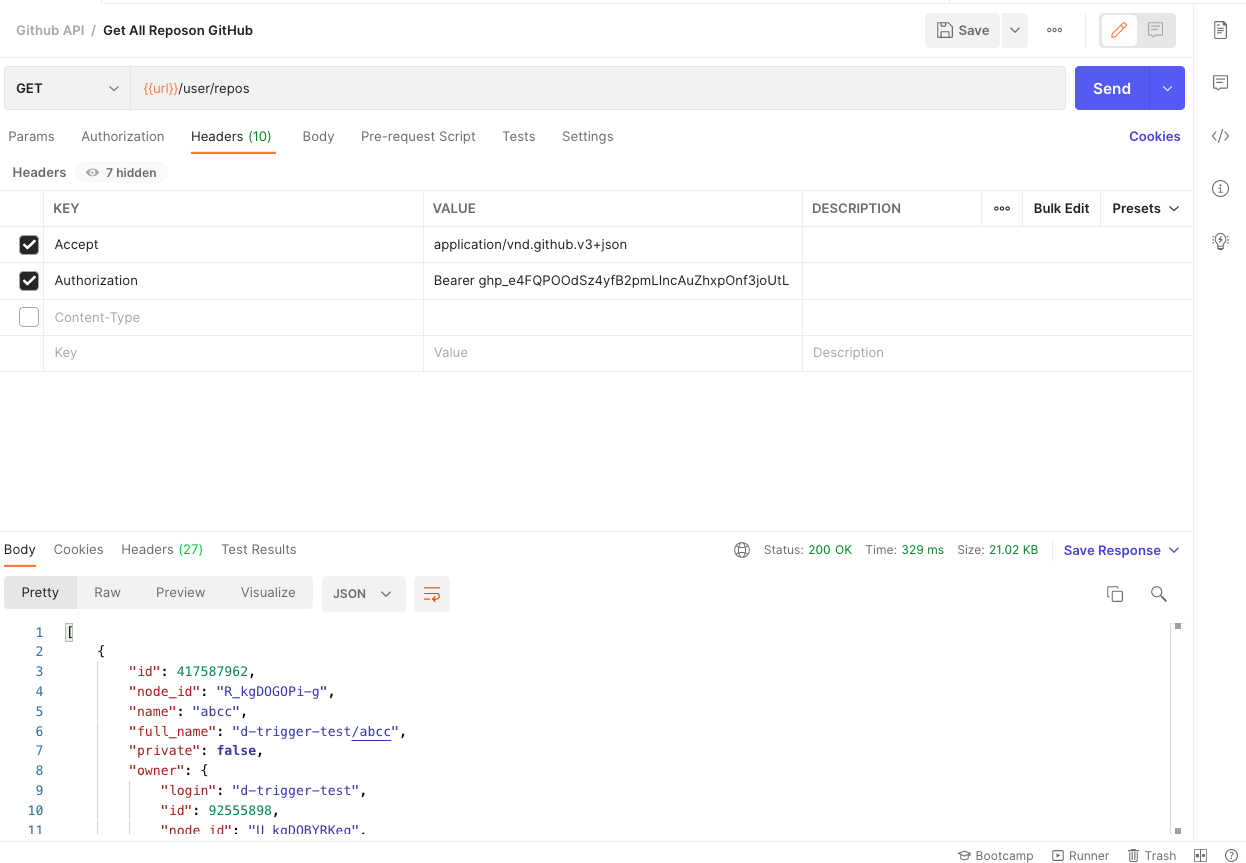
## Manual tests with Postman:

For our tests, I used five different requests that available in GitHub API documentation:

1. Get All Repos
2. Create New Repo
3. Get Exact Repo
4. Edit Repo Name
5. Delete Repo

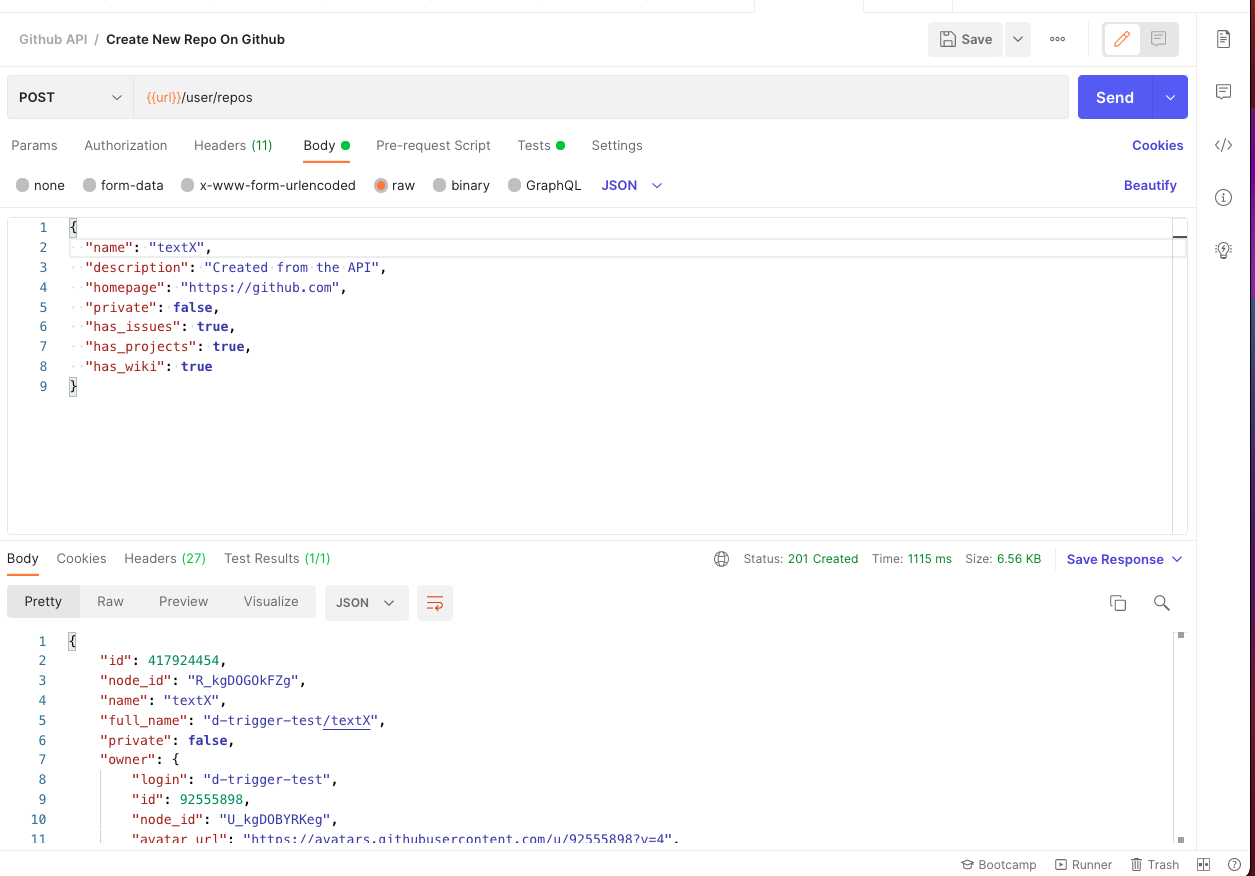
Below we have our requests and JS tests for verification:

1. Get All Repos (pic. 1). Expected status code – 200.

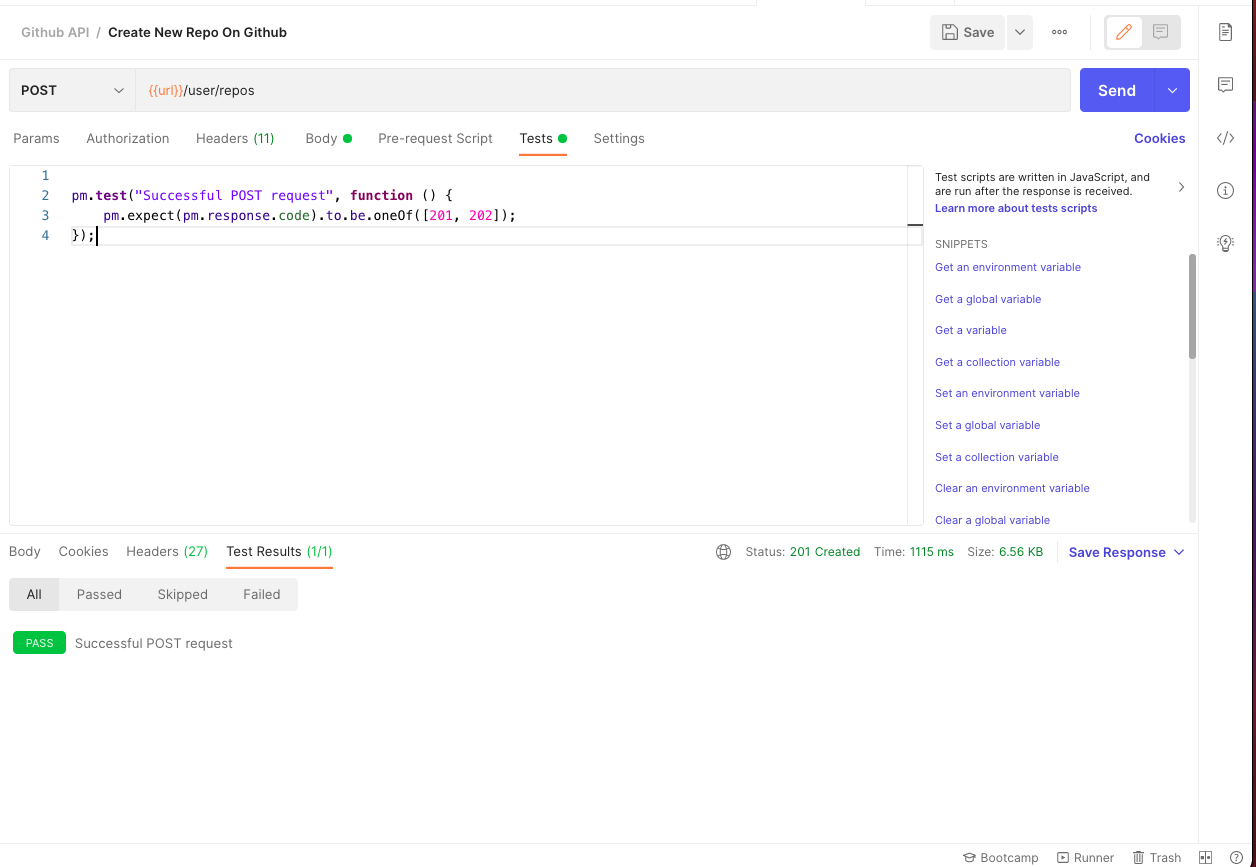


Pic. 1

1. Create New Repo (pic. 2.1) and tests (pic. 2.2). Expected status code – 201.

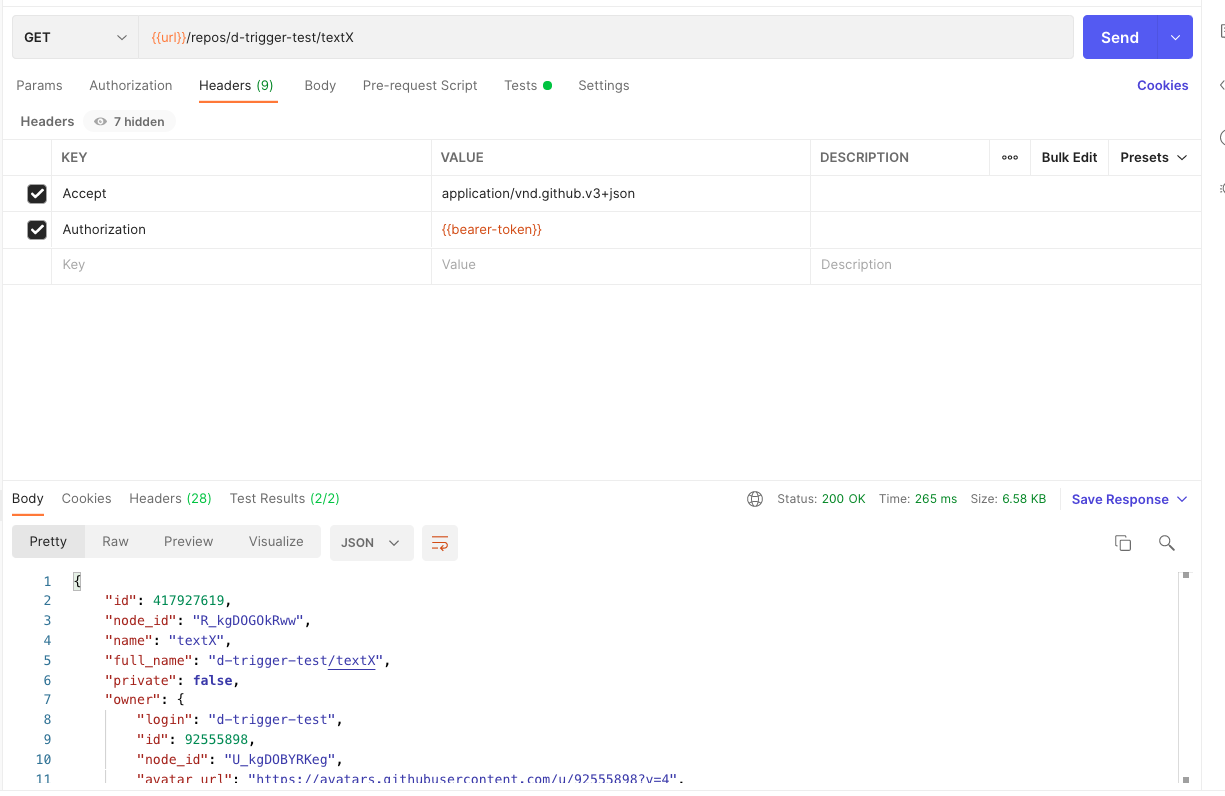


Pic 2.1

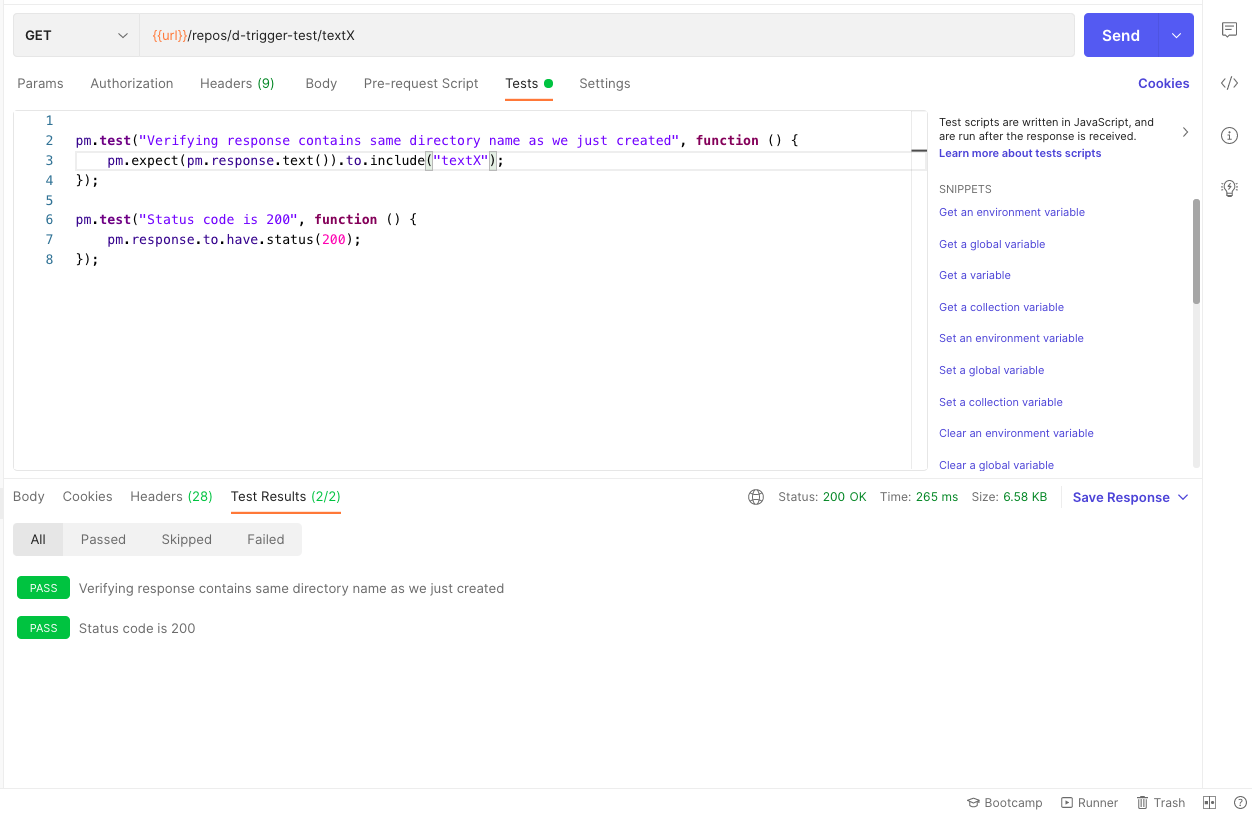


Pic 2.2

1. Get Exact Repo (pic. 3.1) and tests (pic. 3.2). Expected status code – 200.

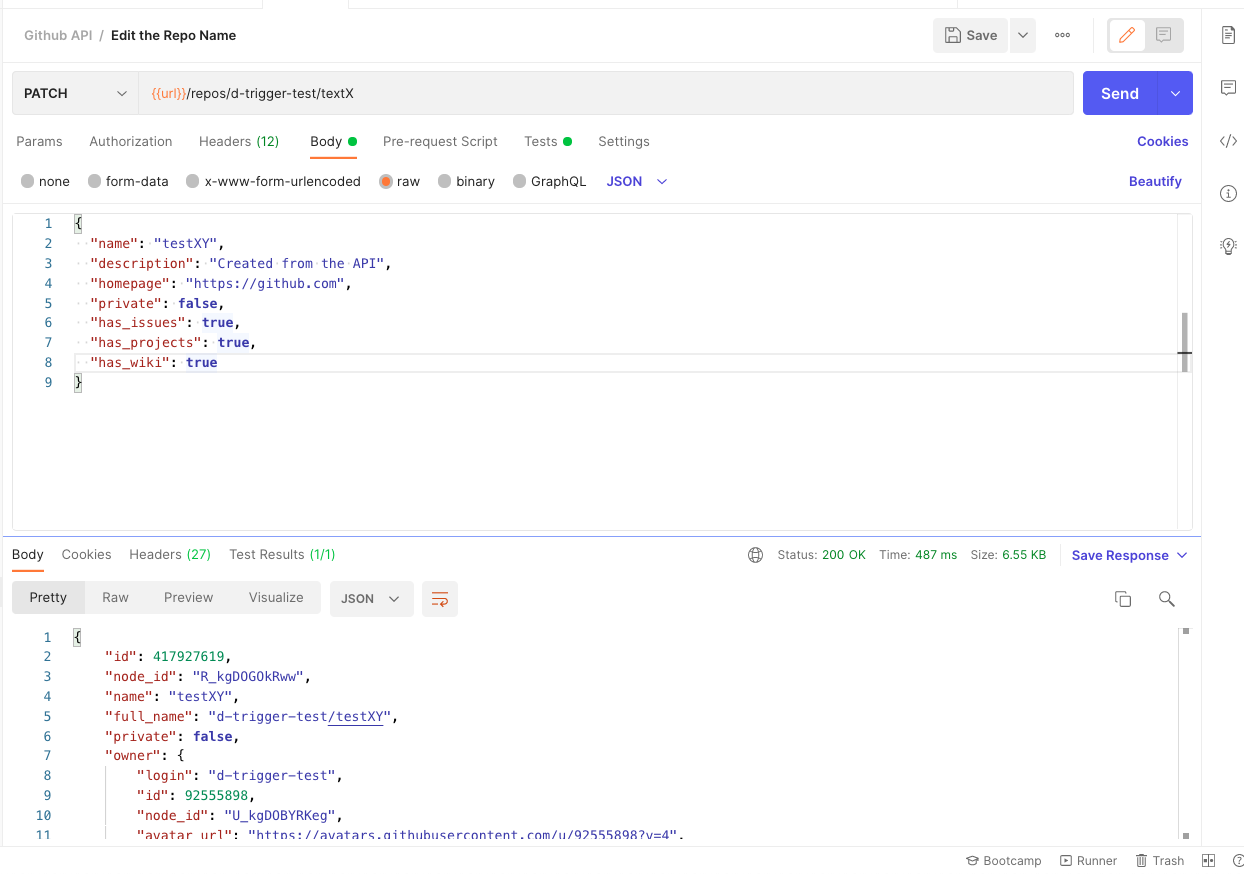


Pic. 3.1

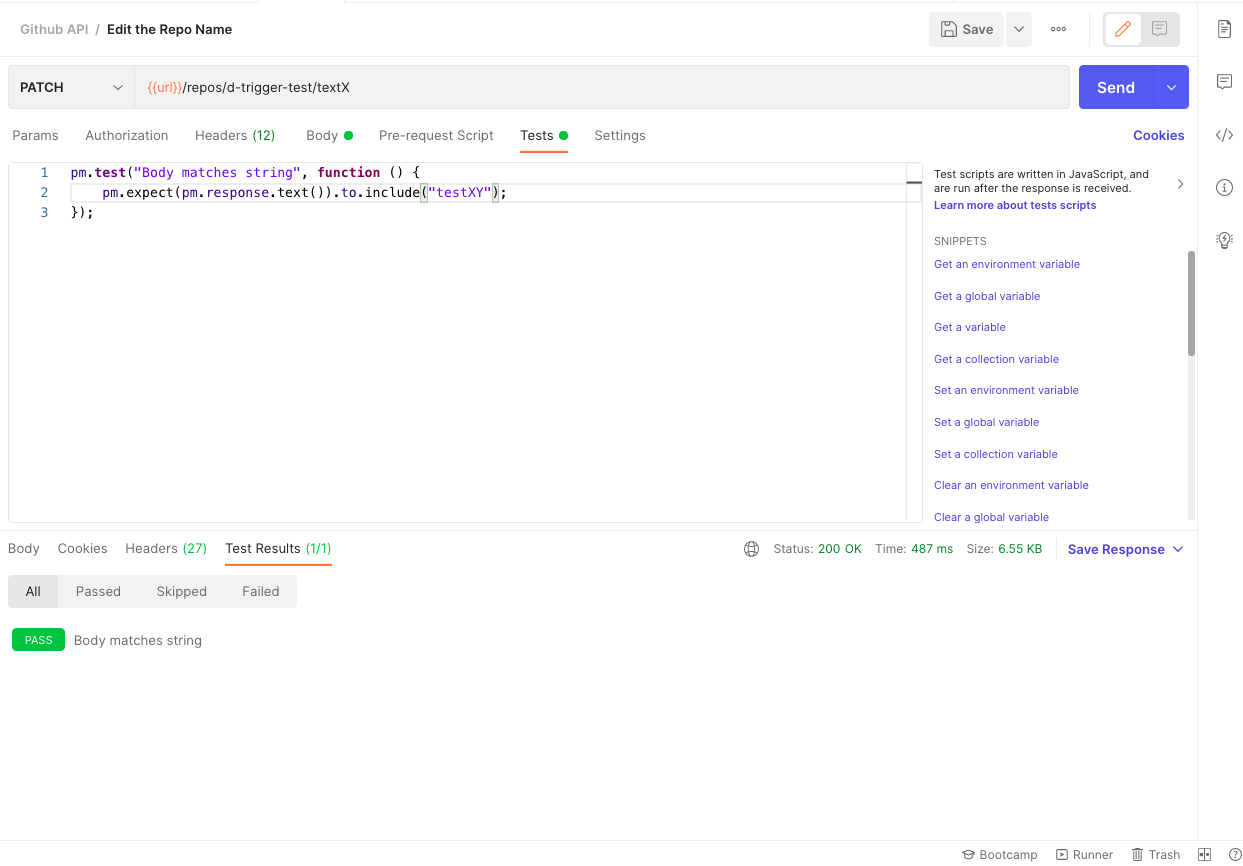


Pic 3.2

1. Edit Repo Name (pic. 4.1) and test (pic. 4.2). Expected status code – 200.



Pic. 4.1



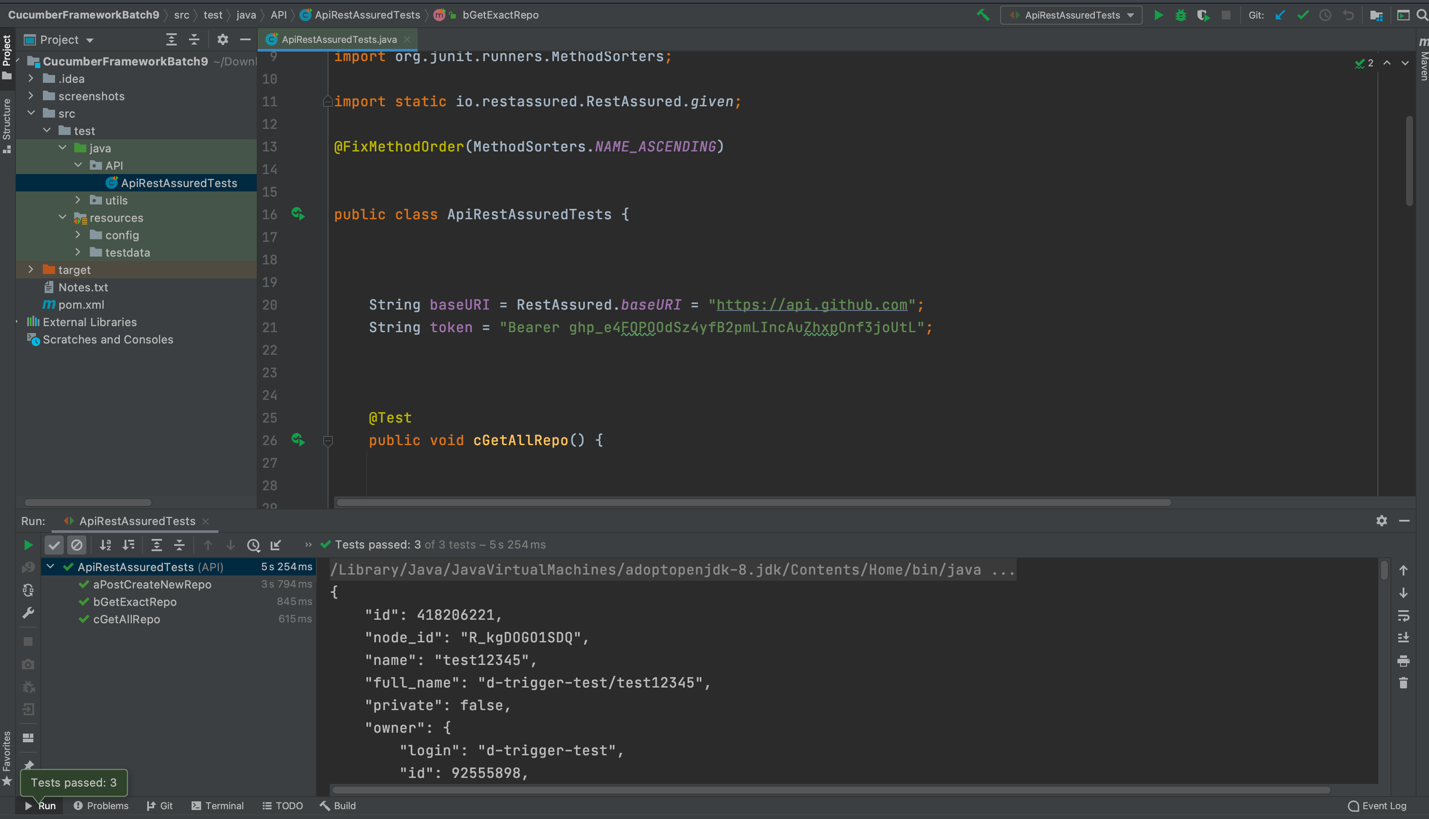
Pic. 4.2

1. Delete Repo (pic. 5). Expected status code – 204.

## 

Pic. 5

## Automated tests in Cucumber framework with JUnit assertions



## 