**Report**

**1. Your evaluation of the quality of the API.**

* Well designed.
* It’s secured to some extent, not allowing user to create new persistent records or update any records permanently. Though users can create, update or delete records (which are temp in database).
* This is achieved by altering the results data structure.
* Cannot handle huge traffic load.

At times when requests are high & server gets busy it returns Error 429.

**2. Details of what you tested.**

* Tested following scenarios: Create user, Get user details, update the existing user & deleting the user.
* Covered both positive & negative paths for every scenario.
* Findings listed in observations section.

**3. Details of how you tested it.**

* Used POSTMAN for manually testing the endpoints.
* RestAssured java library to write automation scripts to test endpoints.

**4. Suggestions as to how you and the team could have done a better job of**

**developing and testing the API.**

* Define standard practices and security guidelines for API development.
* Monitoring the development based on defined guidelines.
* Conducting audits

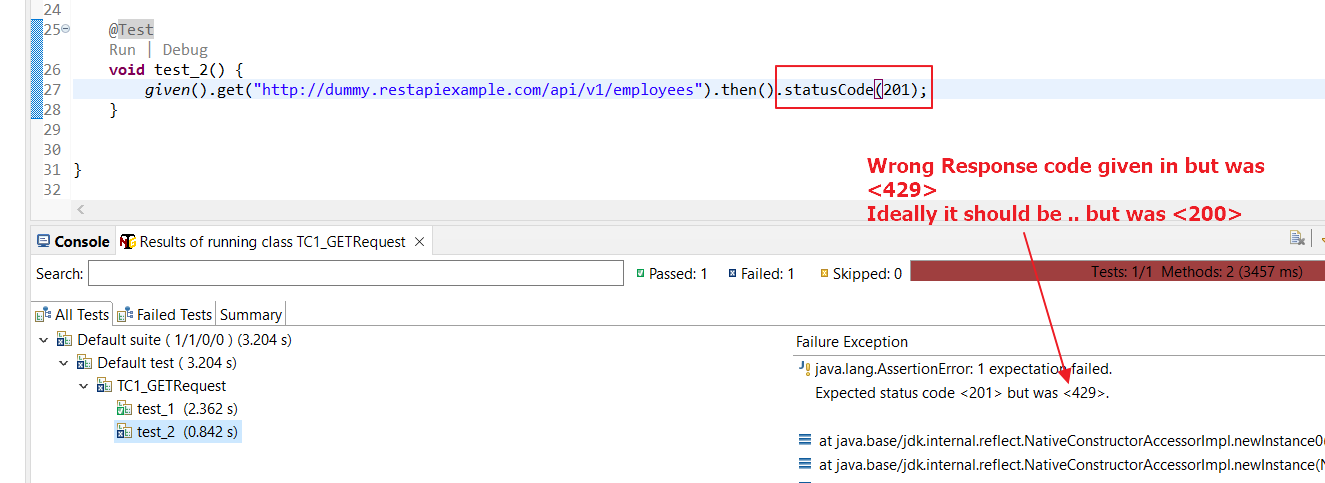
**5. Suggested approaches for the whole team to test this API together (e.g a**

**mobbing session).**

* Brain storming session
* Peer programming, review & testing.
* Introduce early testing for the API under development.
* A mindmap of test scenarios should be drawn before API is developed and pass on to developers.

**Observations:**

Observation 1: Wrong response code. The expected response should have been – Expected status code <201> but was <200>

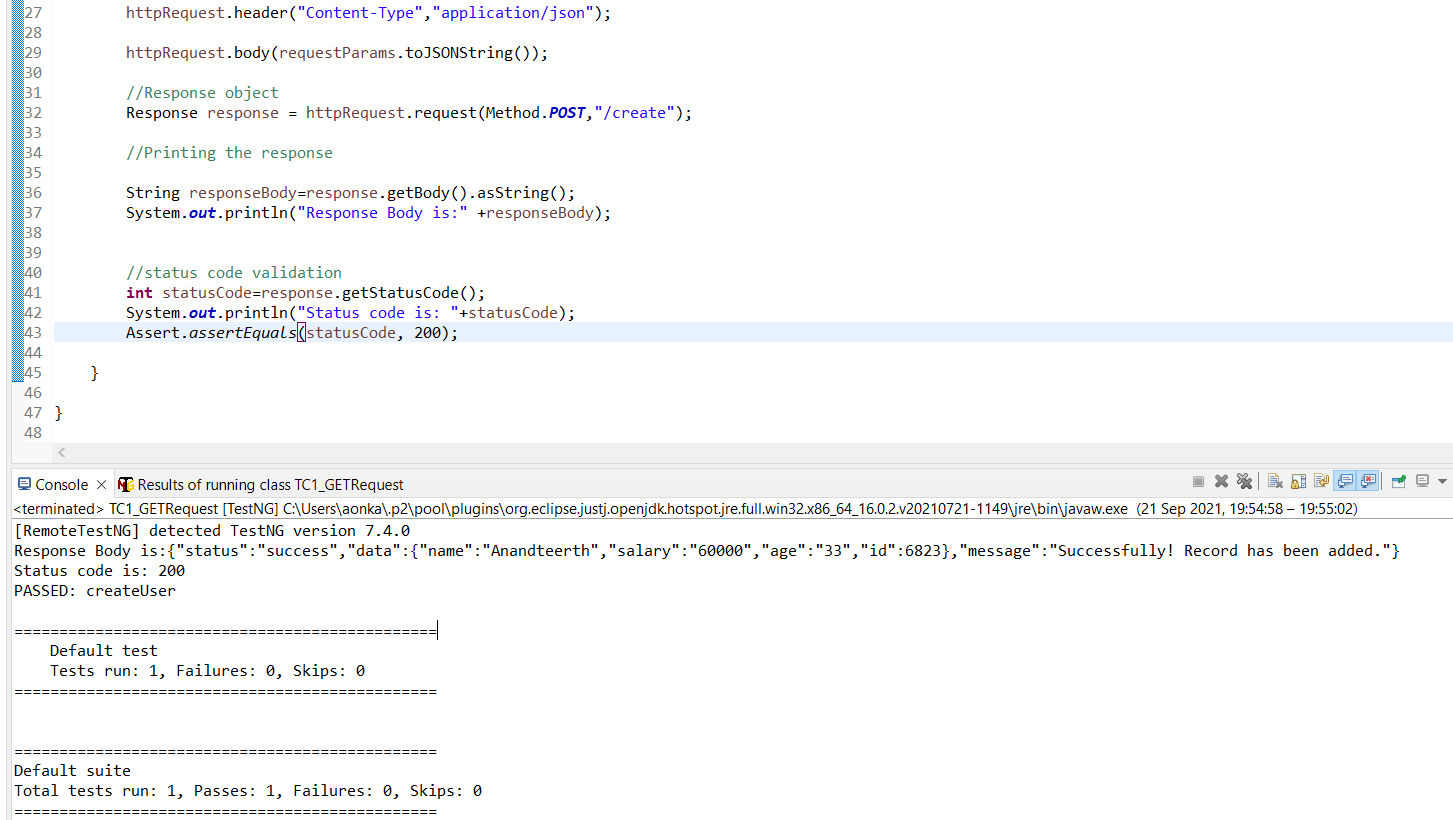


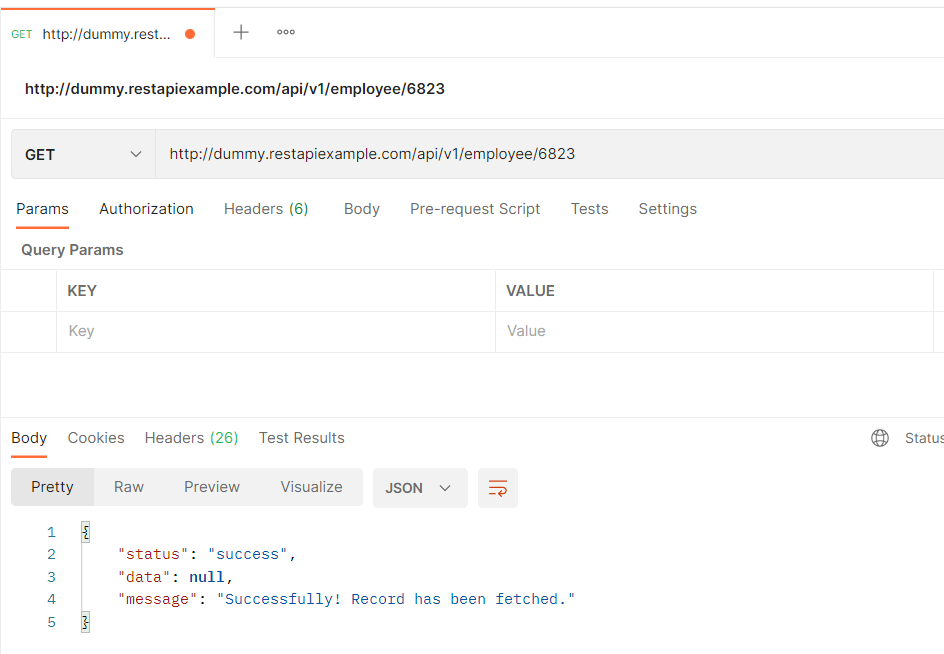
Observation 2: Record is created successful, but status code – 200 & not 201

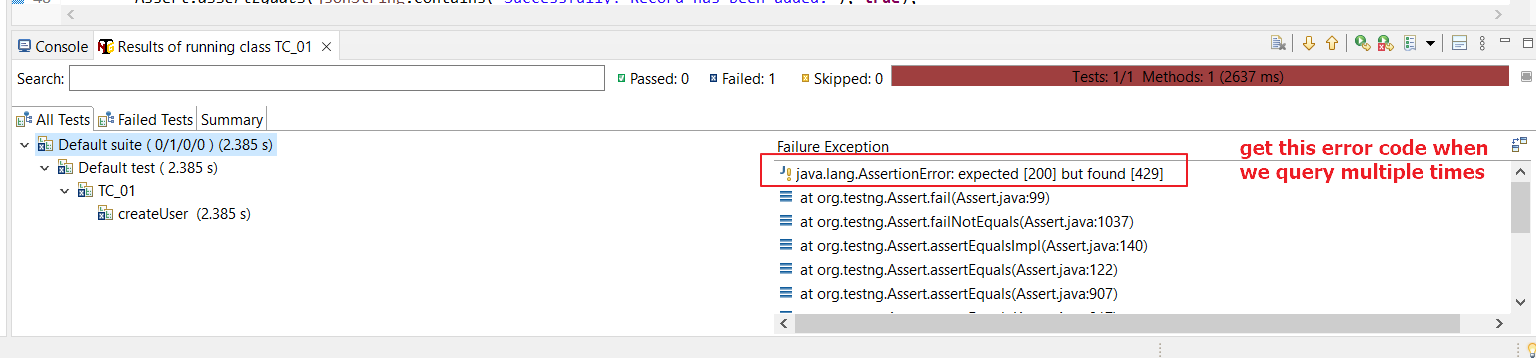
Created record cannot be fetched. Application is designed in that way.



When we try to fetch the created record, the data is shown as Null







Negative Scenario:

