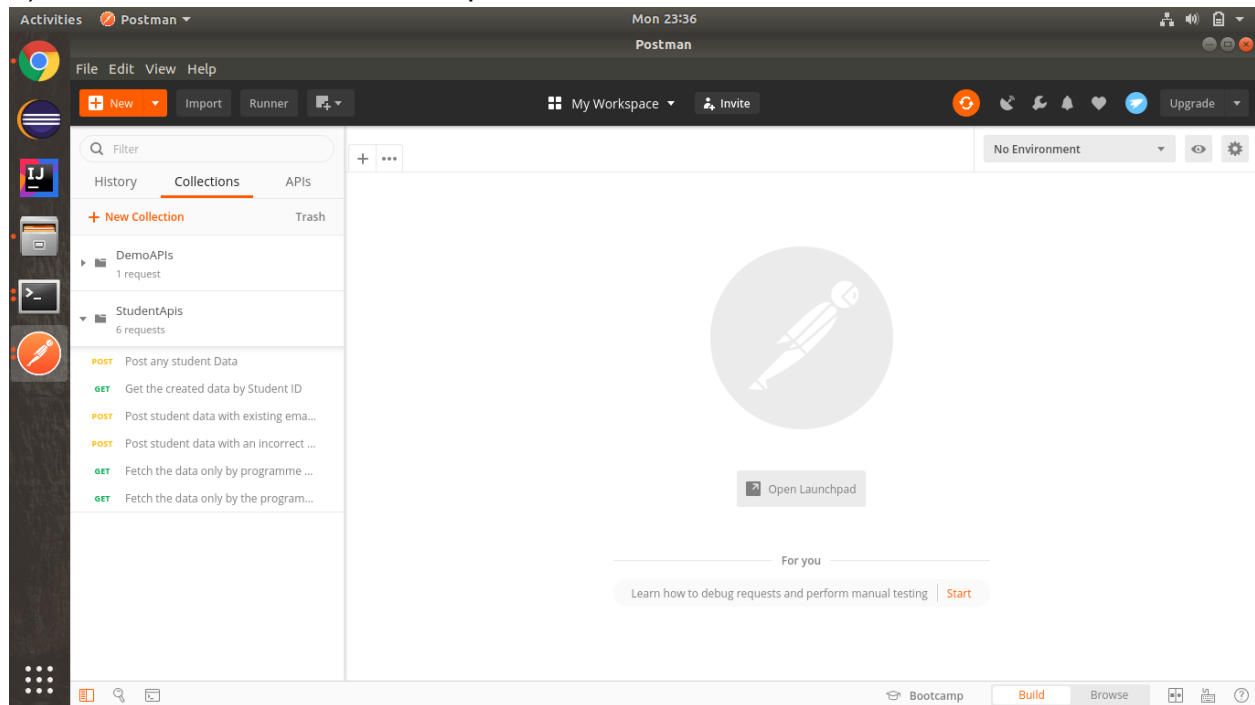


1) Create new collection and map the API's in the same folder

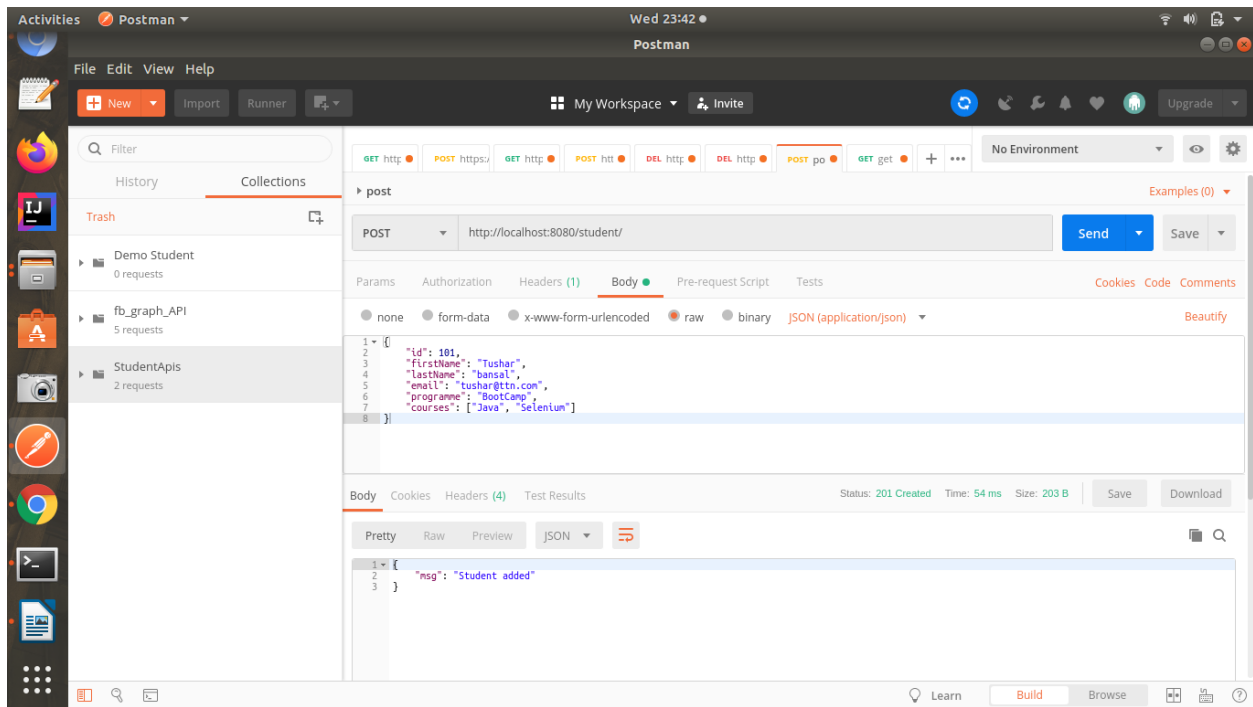


2) Post any student Data and get the created data by Student ID

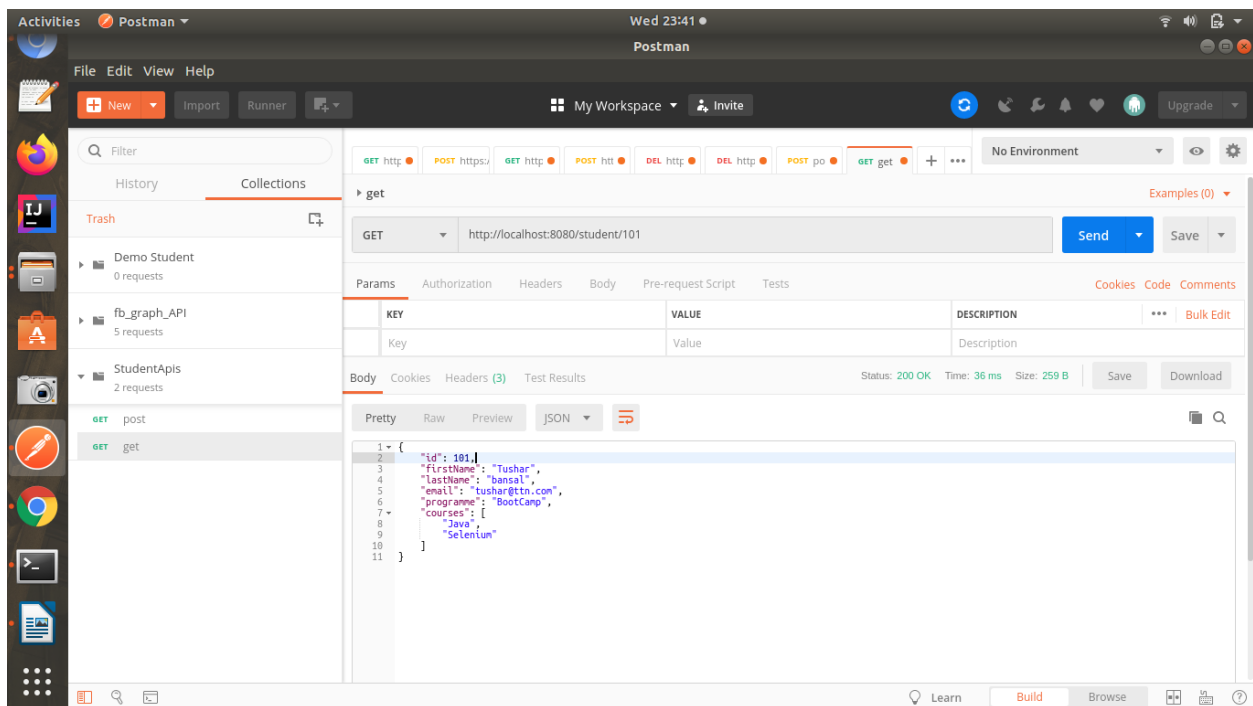
Post : <http://localhost:8080/student/>

Body :

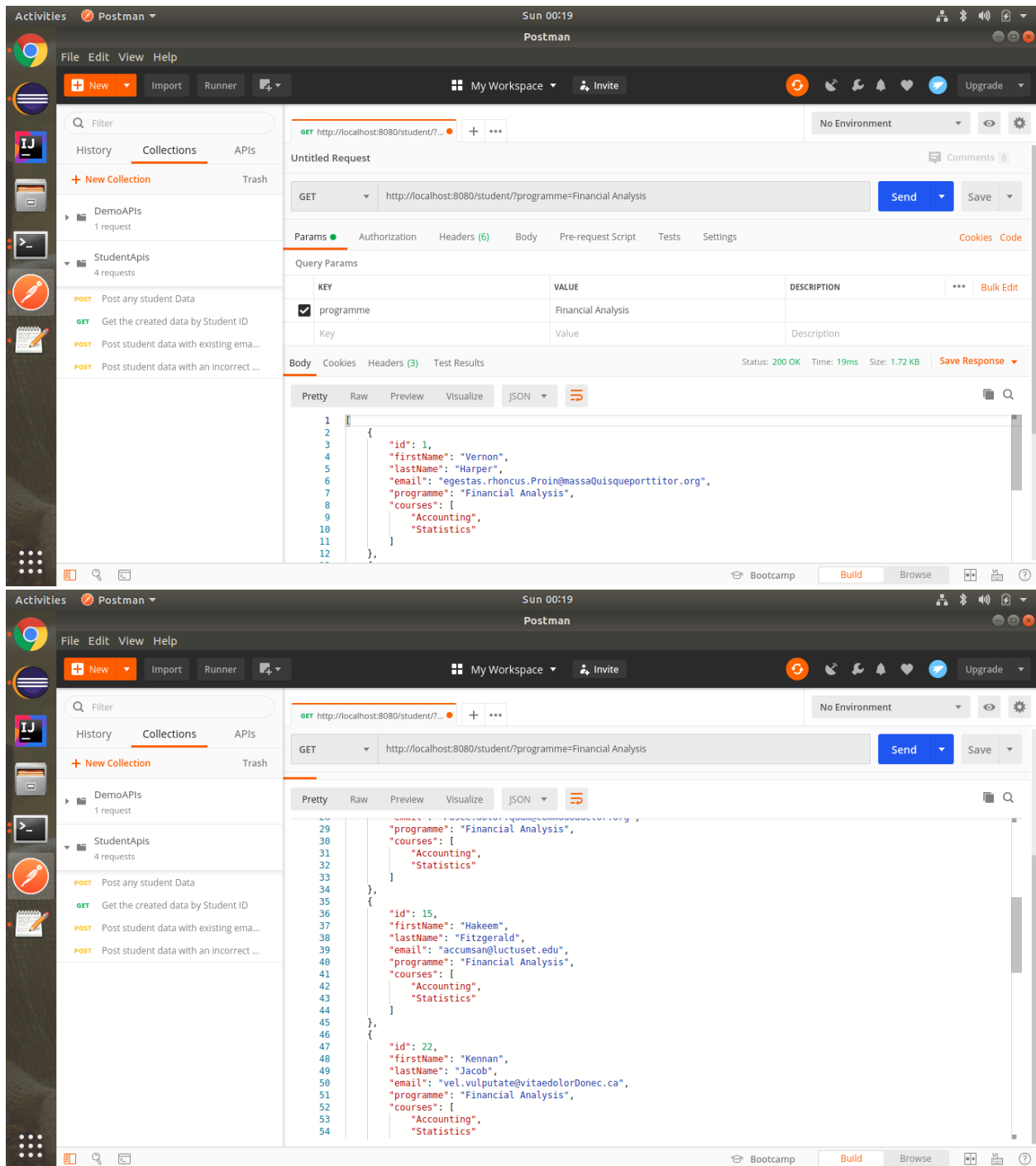
```
{
  "id": 101,
  "firstName": "Tushar",
  "lastName": "Bansal",
  "email": "tushar@ttn.com",
  "programme": "BootCamp",
  "courses": ["Java", "Selenium"]
}
```



Get : <http://localhost:8080/student/101>



3) Fetch the data only by programme "Financial Analysis"
Get : <http://localhost:8080/student/?programme=Financial Analysis>



4) Post student data with existing email Id and check for the error message. Mention the test script.

Post : <http://localhost:8080/student/>

Body :

```

{
  "id": 102,

```

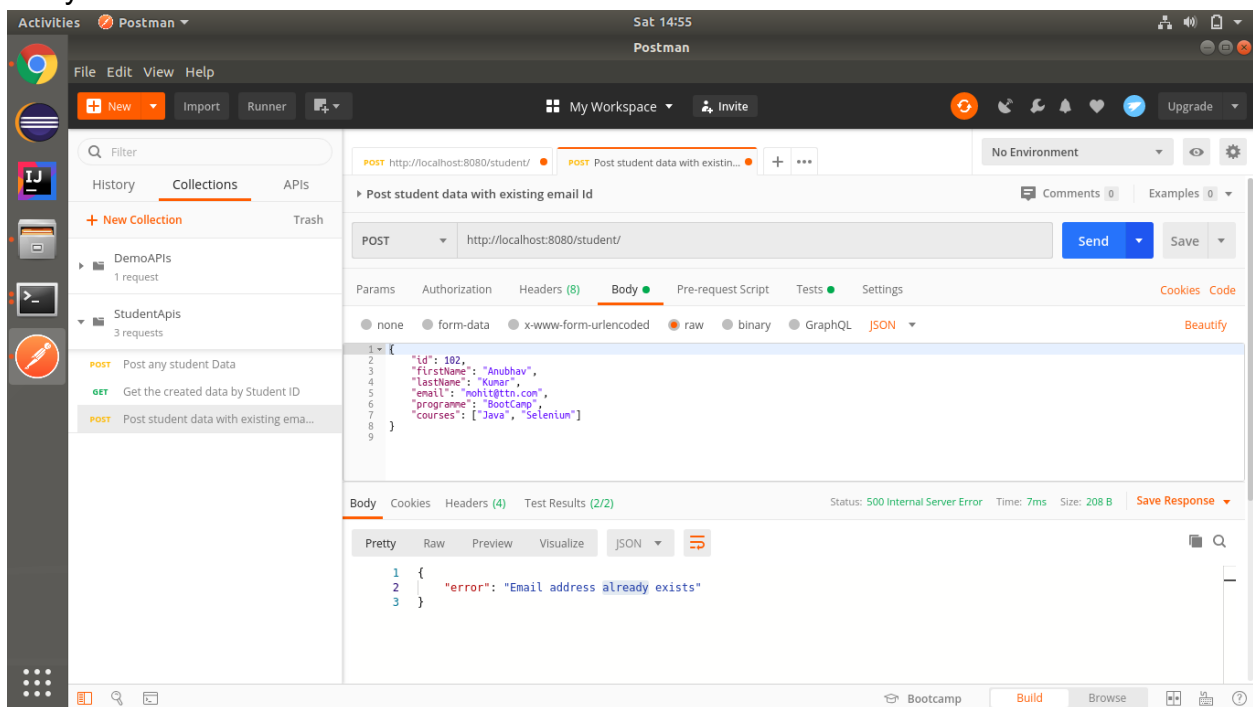
```
"firstName": "Anubhav",  
"lastName": "Kumar",  
"email": "mohit@ttn.com",  
"programme": "BootCamp",  
"courses": ["Java", "Selenium"]  
}
```

Test :

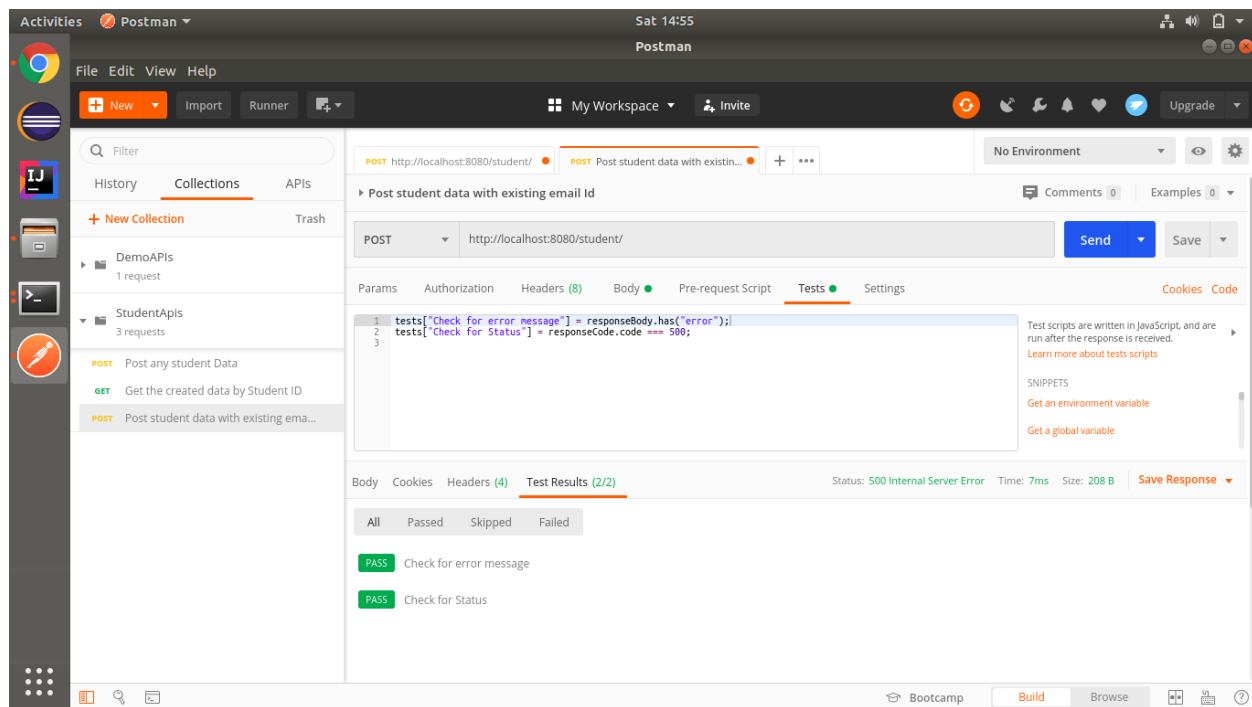
```
tests["Check for error message"] = responseBody.has("error");  
tests["Check for Status"] = responseCode.code === 500;
```

Output :-

Body :



Test :



5) Post student data with an incorrect format of email Id and check for the error message. Mention the test script.

Post : <http://localhost:8080/student/>

Body :

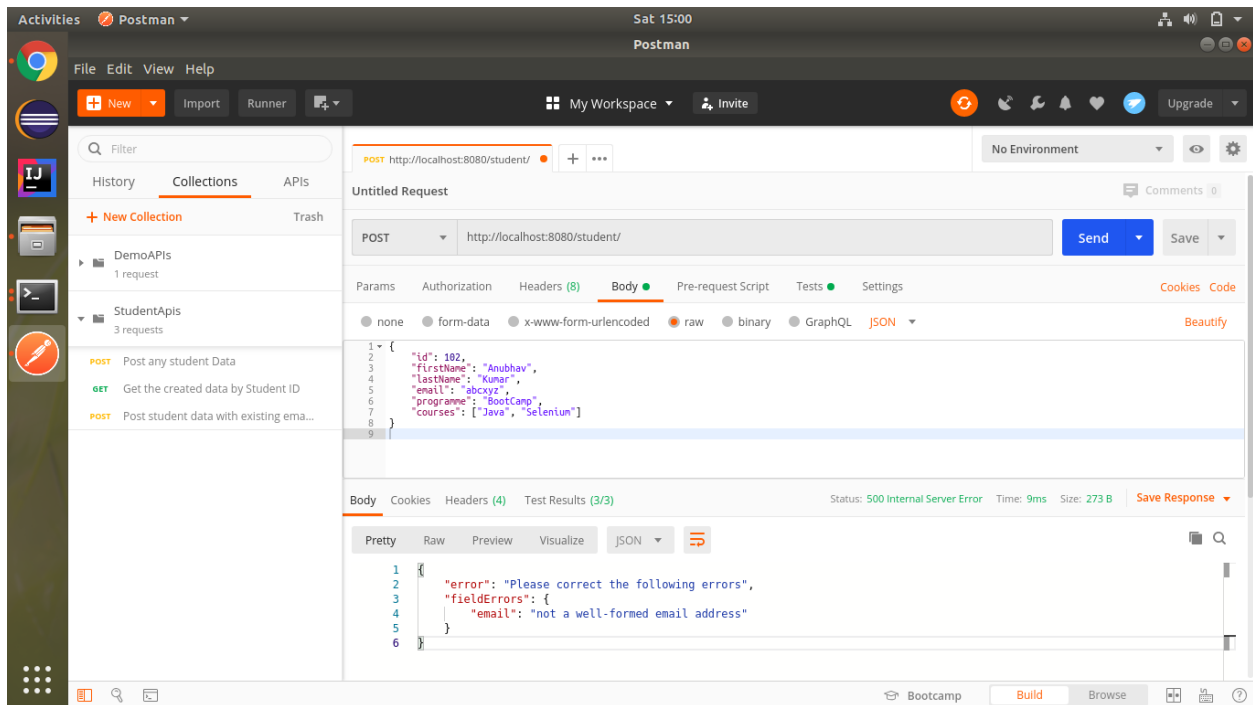
```
{
  "id": 102,
  "firstName": "Anubhav",
  "lastName": "Kumar",
  "email": "abcxyz",
  "programme": "BootCamp",
  "courses": ["Java", "Selenium"]
}
```

Tests:

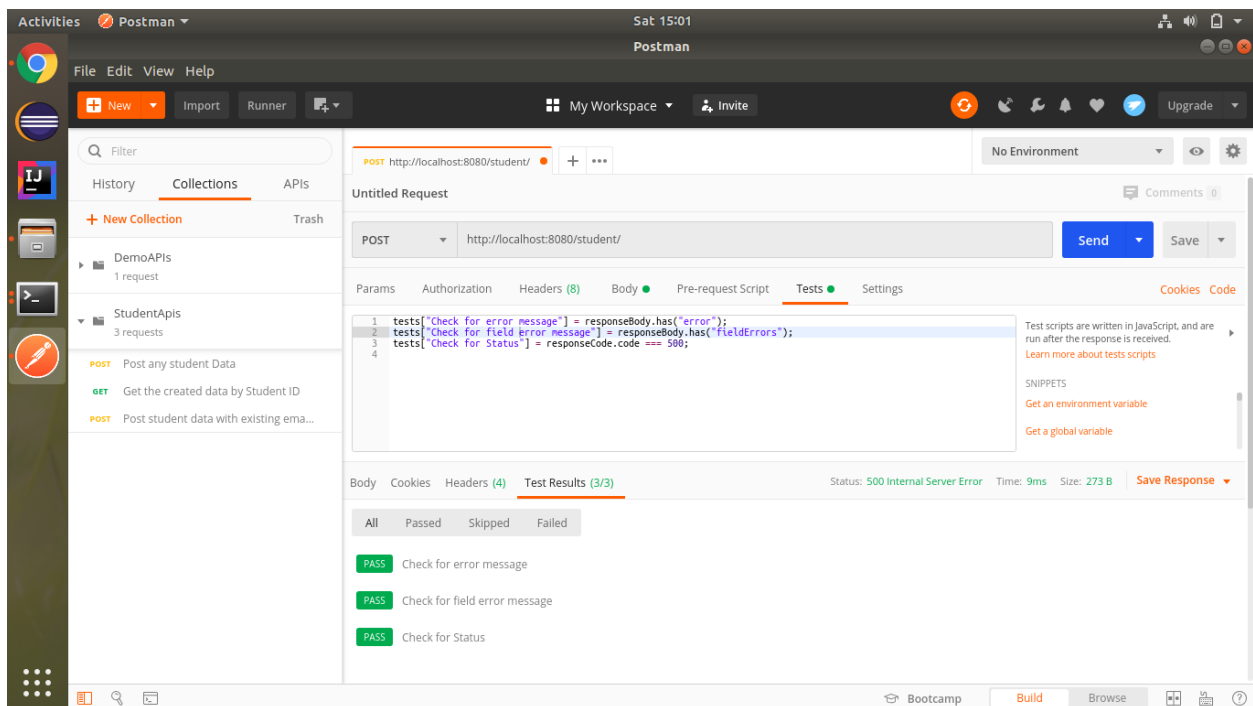
```
tests["Check for error message"] = responseBody.has("error");
tests["Check for field error message"] = responseBody.has("fieldErrors");
tests["Check for Status"] = responseCode.code === 500;
```

Output :-

Body :



Test :



6) Fetch the data only by the programme "Computer Science" and "Criminology".

Mention the test script

Get : `http://localhost:8080/student/?programme=Computer Science&Programme=Criminology`

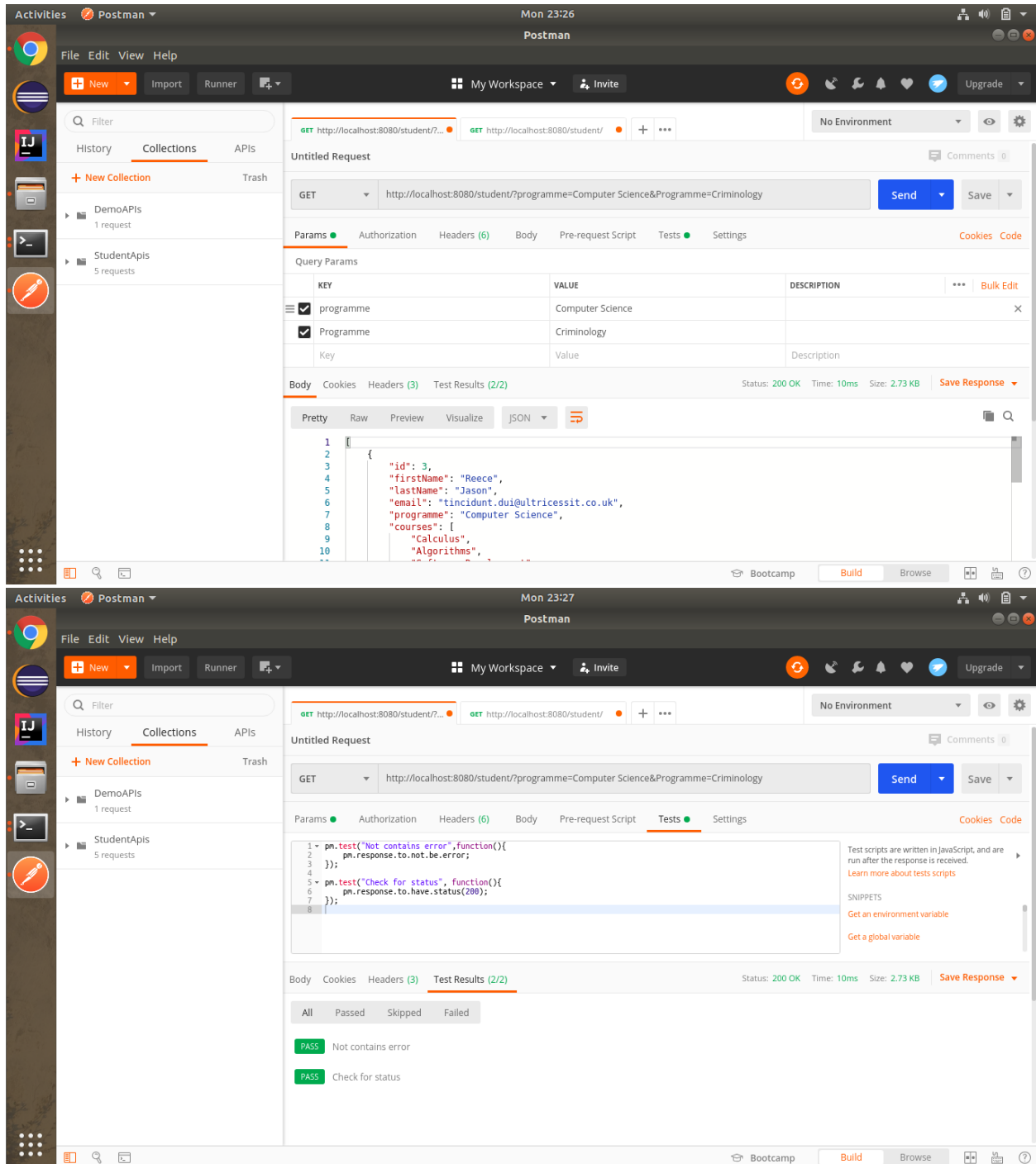
Tests :

`pm.test("Not contains error",function(){`

```

pm.response.to.not.be.error;
});
pm.test("Check for status", function(){
    pm.response.to.have.status(200);
});

```



7) Run all the above cases and generate the report.

Activities Postman Mon 23:33 Collection Runner

File Edit View Help

Collection Runner Run Results My Workspace Run In Command Line Docs

7 PASSED 0 FAILED StudentApis No Environment just now Run Summary Export Results Retry New

Iteration 1

- POST Post any student Data http://localhost:8080/stu... ..udentApis / Post any student Data 201 Created 353 ms 203 B
This request does not have any tests.
- GET Get the created data by Student ID http://localhost:8080/stu... ..et the created data by Student ID 200 OK 66 ms 256 B
This request does not have any tests.
- POST Post student data with existing email Id http://localhost:8080/stu... ..tudent data with existing email Id 500 Internal Server Error 36 ms 208 B
 - Check for error message
 - Check for Status
- POST Post student data with an incorrect format of email Id http://localhost:8080/stu... ..lth an incorrect format of email Id 500 Internal Server Error 77 ms 273 B
 - Check for error message
 - Check for field error message
 - Check for Status
- GET Fetch the data only by programme "Financial Analysis" http://localhost:8080/stu... ..y programme "Financial Analysis" 200 OK 69 ms 1.76 KB
This request does not have any tests.

Activities Postman Mon 23:34 Collection Runner

File Edit View Help

Collection Runner Run Results Run Summary My Workspace Run In Command Line Docs

7 PASSED 0 FAILED StudentApis No Environment just now Export Results New

Back 1

- POST Post any student Data
- GET Get the created data by Student ID
- POST Post student data with existing email Id ✓
- POST Post student data with an incorrect format ... ✓
- GET Fetch the data only by programme "Financi...
- GET Fetch the data only by the programme "Co..." ✓