Mocha and Chai Unit Testing

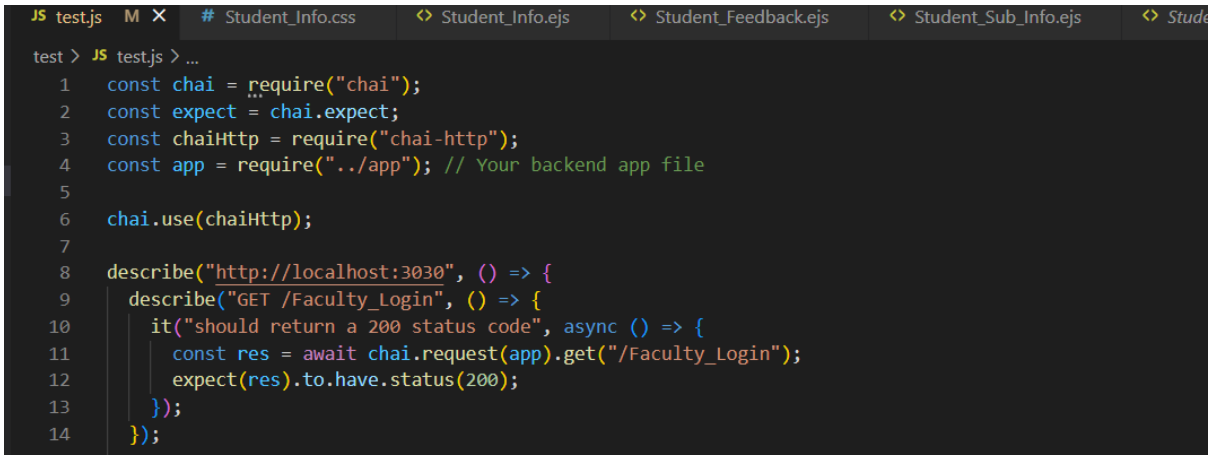
Isha Popat: 202001095

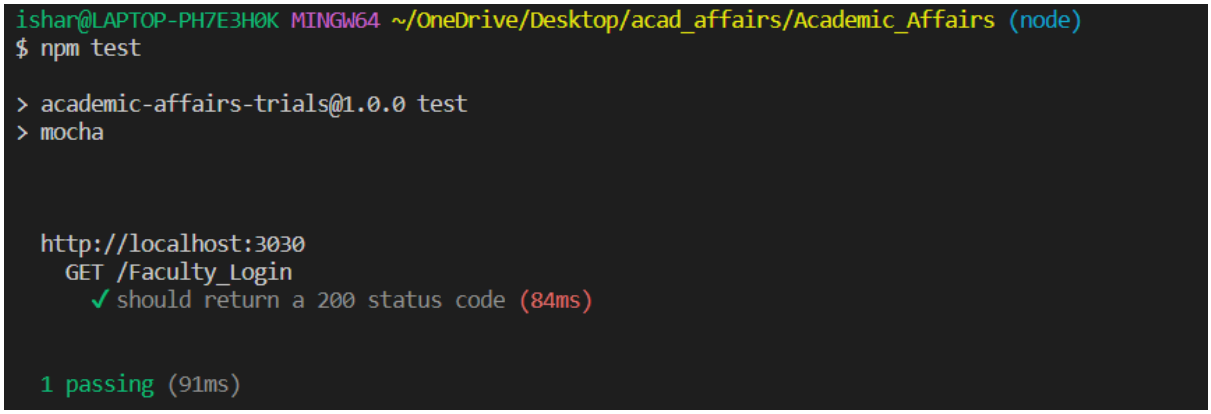
Kush Shah : 202001104

1) For the first test case:

When the expected status code matches the status code returned and hence this test passes.

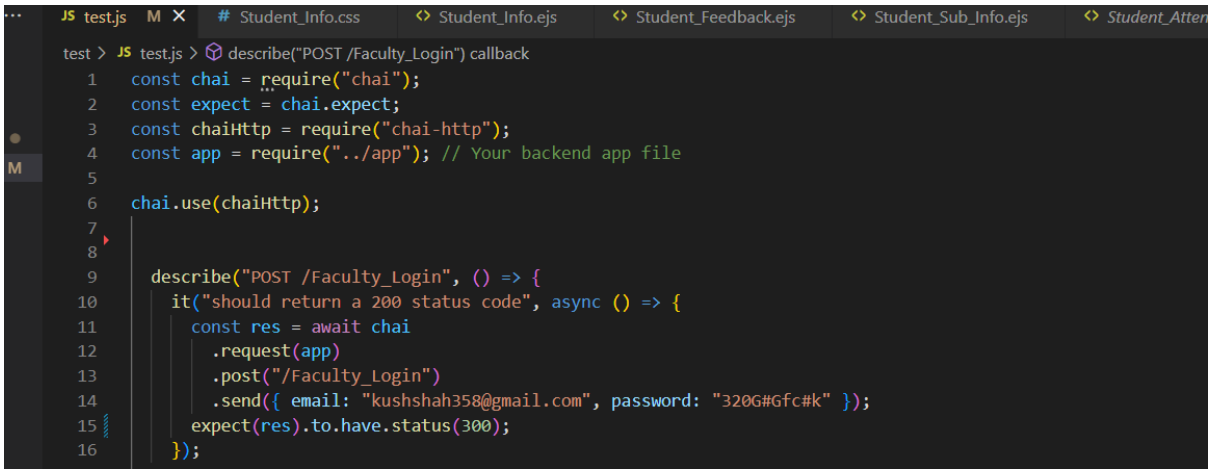
This is when the status code of the faculty login matches with the expected status code





2) For the second test case:

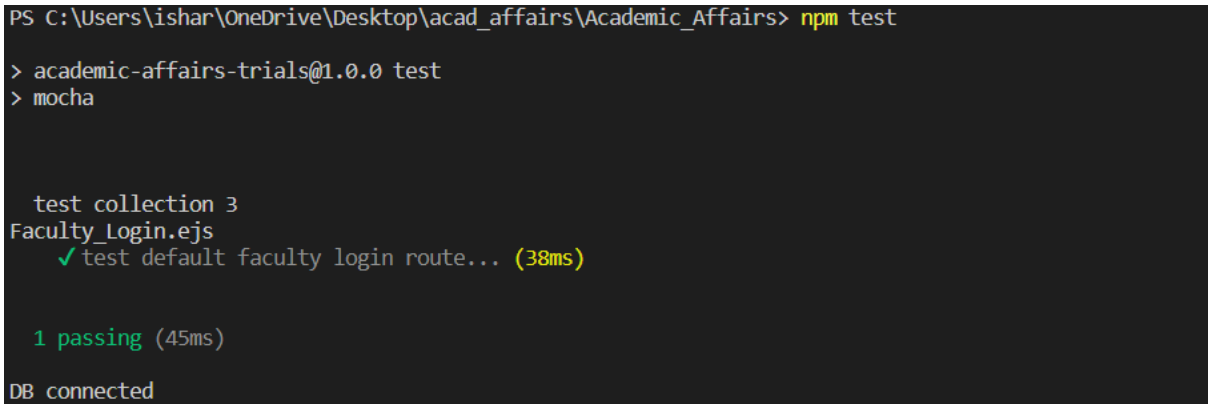
when the expected status code does not match the status code returned and hence this test passes




```
ishar@LAPTOP-PH7E3H0K MINGW64 ~/OneDrive/Desktop/acad_affairs/Academic_Affairs (node)
$ npm test

> academic-affairs-trials@1.0.0 test
(node:3716) [DEP0066] DeprecationWarning: OutgoingMessage.prototype._headers is deprecated
(Use `node --trace-deprecation ...` to show where the warning was created)
    1) should return a 200 status code


  0 passing (120ms)
  1 failing

  1) POST /Faculty_Login
       should return a 200 status code:

     AssertionError: expected { Object (_events, _eventsCount, ...) } to have status code 300 but got 200
     + expected - actual

     -200
     +300

     at Context.<anonymous> (test\test.js:15:27)
     at process.processTicksAndRejections (node:internal/process/task_queues:95:5)
```

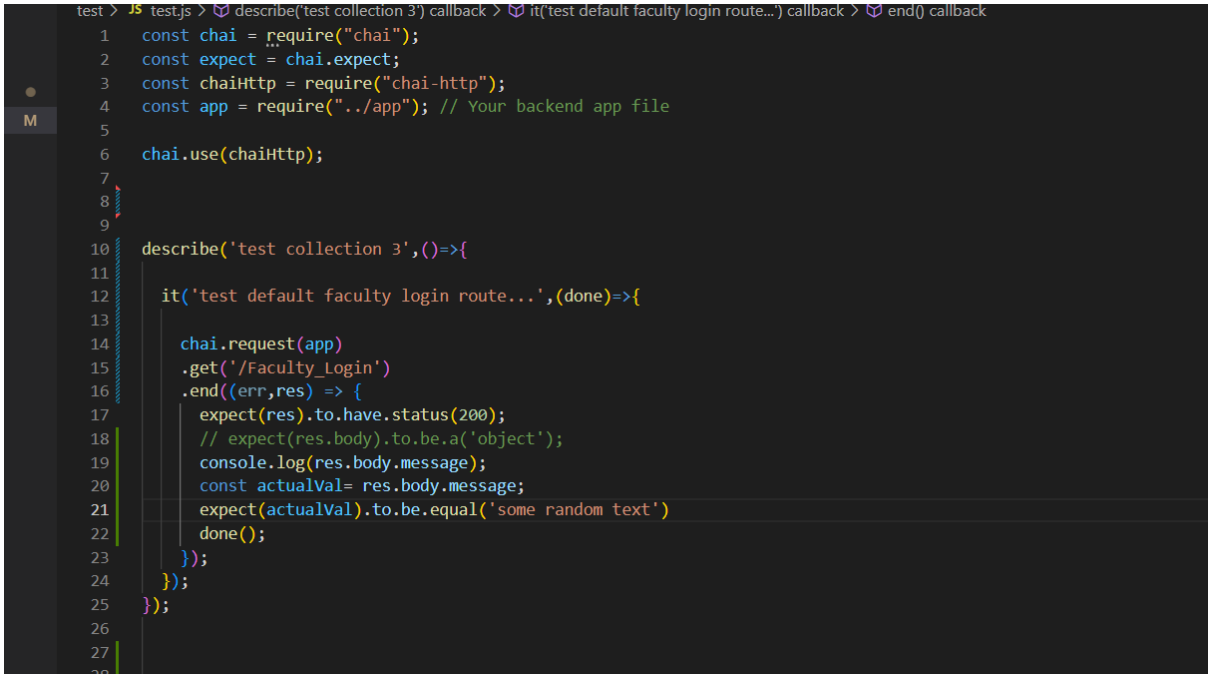
3) for this test case the render message at status 200 is ‘Faculty_Login.ejs’ and the expected message is ‘Faculty_Login.ejs ‘ so it matches and hence the test case passes.

```
test > JS test.js > ...
1   const chai = require("chai");
2   const expect = chai.expect;
3   const chaiHttp = require("chai-http");
4   const app = require("../app"); // Your backend app file
5
6   chai.use(chaiHttp);
7
8
9
10  describe('test collection 3',()=>{
11
12    it('test default faculty login route...',(done)=>{
13
14      chai.request(app)
15      .get('/Faculty_Login')
16      .end((err,res) => {
17        expect(res).to.have.status(200);
18        // expect(res.body).to.be.a('object');
19        console.log(res.body.message);
20        const actualVal= res.body.message;
21        expect(actualVal).to.be.equal('Faculty_Login.ejs')
22        done();
23      });
24    });
25  });
26
```



```
PS C:\Users\ishar\OneDrive\Desktop\acad_affairs\Academic_Affairs> npm test

> academic-affairs-trials@1.0.0 test
> mocha



  test collection 3
Faculty_Login.ejs
    ✓ test default faculty login route... (38ms)


  1 passing (45ms)

DB connected
```

4) for this test case the render message at status 200 is ‘Faculty_Login.ejs’ but the expected message is ‘some random text’ so it doesn’t match and hence fails.



```
test > JS test.js > describe('test collection 3') callback > it('test default faculty login route...') callback > end() callback
1  const chai = require("chai");
2  const expect = chai.expect;
3  const chaiHttp = require("chai-http");
4  const app = require("../app"); // Your backend app file
5
6  chai.use(chaiHttp);
7
8
9
10 describe('test collection 3',()=>{
11
12   it('test default faculty login route...',(done)=>{
13
14     chai.request(app)
15     .get('/Faculty_Login')
16     .end((err,res) => {
17       expect(res).to.have.status(200);
18       // expect(res.body).to.be.a('object');
19       console.log(res.body.message);
20       const actualVal= res.body.message;
21       expect(actualVal).to.be.equal('some random text')
22       done();
23     });
24   });
25 });
26
27
```

```
> academic-affairs-trials@1.0.0 test
> mocha



  test collection 3
Faculty_Login.ejs
    1) test default faculty login route...


  0 passing (117ms)
  1 failing

  1) test collection 3
       test default faculty login route...:

      Uncaught AssertionError: expected 'Faculty_Login.ejs' to equal 'some random text'
      + expected - actual

      -Faculty_Login.ejs
      +some random text

      at C:\Users\ishar\OneDrive\Desktop\acad_affairs\Academic_Affairs\test\test.js:21:31
      at Request.callback (node_modules\superagent\lib\node\index.js:716:12)
      at C:\Users\ishar\OneDrive\Desktop\acad_affairs\Academic_Affairs\node_modules\superagent\lib\node\index.js:916:18
      at IncomingMessage.<anonymous> (node_modules\superagent\lib\node\parsers\json.js:19:7)
```

since we used passport and used it for authentication , mocha and chai wont allow us to normally get what user were authenticated as it's a complex procedure . Hence we were only able to test via this in login functionality