Testing:

The Things That I tested in my main app.js file
This file provides data on successful login page we should redirect faculty to their home page by sending status code 200 this is for my post request

```javascript
    var mailOptions = {
      from: "202001104@daiict.ac.in",
      to: req.body.email,
      subject: "Student Information System",
      text: `Your account has been created! Your password is ${randomPass}`,
    };

    transporter.sendMail(mailOptions, function (error, info) {
      if (error) {
        console.log(error);
      } else {
        console.log(req.user + "\n" + req.body.email);
        console.log("Email sent: " + info.response);
      }
    });

    res.status(200).redirect("/Faculty_Login");
  });
  }
})
.catch((err) => {
  console.log("Error:", err);
});
```

For get Request

```javascript
app.get("/Faculty_Login", checkNotAuthenticatedFaculty, function (req, res) {
  res.status(200).render("Faculty_Login.ejs");
});
```

Simple check for rendering page without any external errors
Code for Test.js

Academic-Affairs trials > test > (); test.js > ...

```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    describe("http://localhost:3030", () => {
9      describe("GET /Faculty_Login", () => {
10       it("should return a 200 status code", async () => {
11         const res = await chai.request(app).get("/Faculty_Login");
12         expect(res).to.have.status(200);
13       });
14     });
15
16     describe("POST /Faculty_Login", () => {
17       it("should return a 200 status code", async () => {
18         const res = await chai
19           .request(app)
20           .post("/Faculty_Login")
21           .send({ email: "kushshah358@gmail.com", password: "123" });
22         expect(res).to.have.status(200);
23       });
24
25       it("should return a 200 status code", async () => {
26         const res = await chai
27           .request(app)
28           .post("/Faculty_Login")
29           .send({ email: "kushshah358@gmail.com", password: "1234" });
30         expect(res).to.have.status(200);
31       });
32     });
33   });
```

Results:

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



  http://localhost:3030
    GET /Faculty_Login
      ✓ should return a 200 status code (57ms)
    POST /Faculty_Login
(node:6472) [DEP0066] DeprecationWarning: OutgoingMessage.prototype._headers is deprecated
(Use `node --trace-deprecation ...` to show where the warning was created)
      ✓ should return a 200 status code
      ✓ should return a 200 status code


  3 passing (158ms)
```

Passed all 3 of our tests