

Module 2 Lab 1

Brandon Trinkle

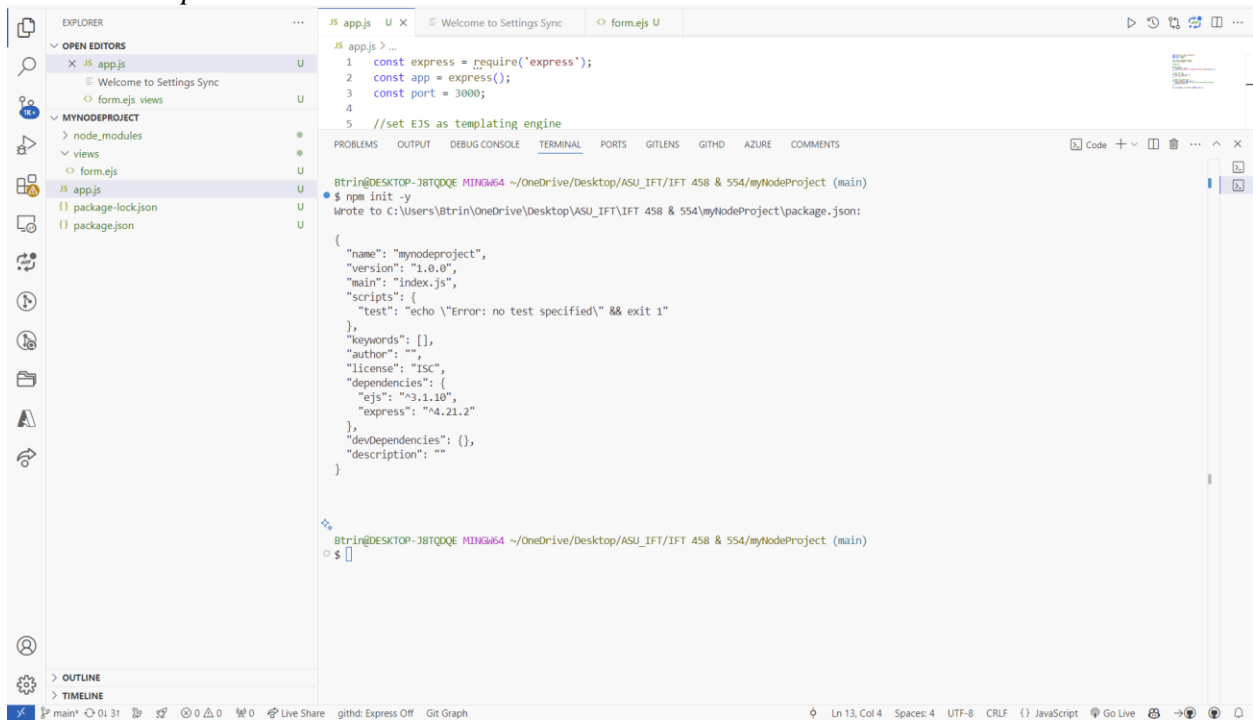
IFT 458: Middleware Prog & Database Sec

Professor: Dinesh Sthapit

Feb 1, 2025

Module 2 Lab 1

1. Terminal displaying the initialized Node.js project, showing the `npm init -y` command and its output.



The screenshot shows the VS Code interface with a terminal window open. The terminal displays the output of the `npm init -y` command, which has created a `package.json` file for the project. The output shows the command being executed and the resulting JSON structure for `package.json`.

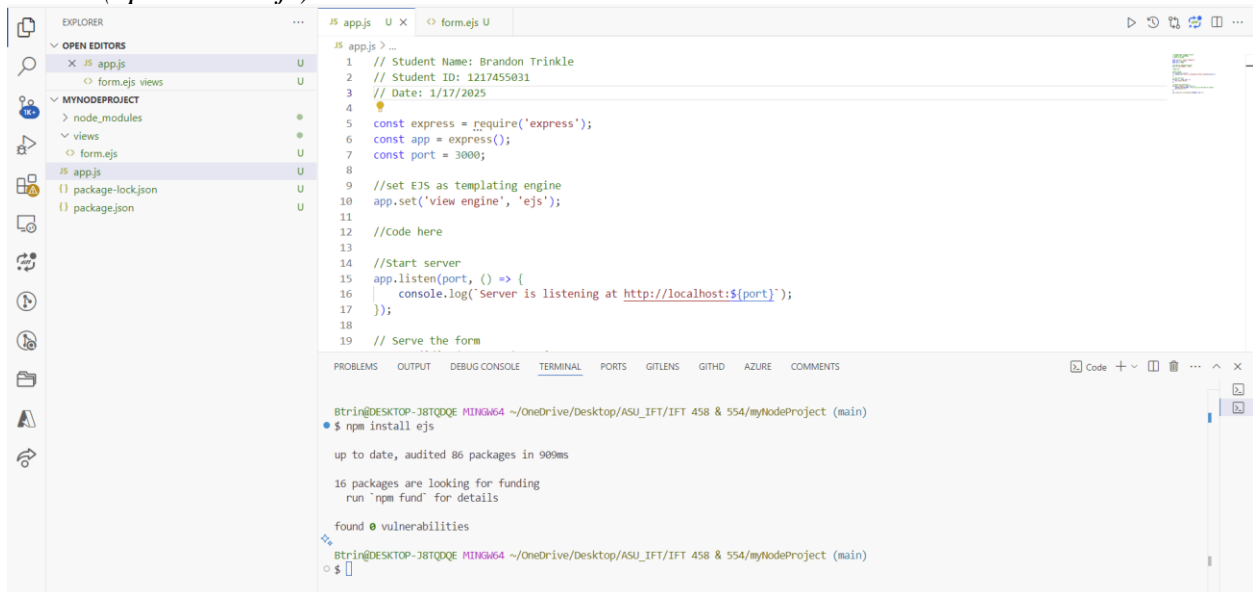
```

Btr@DESKTOP-J8TQDQE MINGW64 ~/OneDrive/Desktop/ASU_IFT/IFT 458 & 554/myNodeProject (main)
$ npm init -y
Wrote to C:\Users\Btr\OneDrive\Desktop\ASU_IFT\IFT 458 & 554\myNodeProject\package.json:

{
  "name": "myNodeProject",
  "version": "1.0.0",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "keywords": [],
  "author": "",
  "license": "ISC",
  "dependencies": {
    "ejs": "^3.1.10",
    "express": "^4.21.2"
  },
  "devDependencies": {},
  "description": ""
}

Btr@DESKTOP-J8TQDQE MINGW64 ~/OneDrive/Desktop/ASU_IFT/IFT 458 & 554/myNodeProject (main)
$
  
```

2. Terminal showing the successful installation of Express.js (`npm install express`) and EJS (`npm install ejs`).



The screenshot shows the VS Code interface with a terminal window open. The terminal displays the output of the `npm install express` and `npm install ejs` commands. The output shows the packages being installed and the resulting JSON structure for `package.json`.

```

Btr@DESKTOP-J8TQDQE MINGW64 ~/OneDrive/Desktop/ASU_IFT/IFT 458 & 554/myNodeProject (main)
$ npm install express
up to date, audited 86 packages in 909ms

16 packages are looking for funding
run 'npm fund' for details

found 0 vulnerabilities

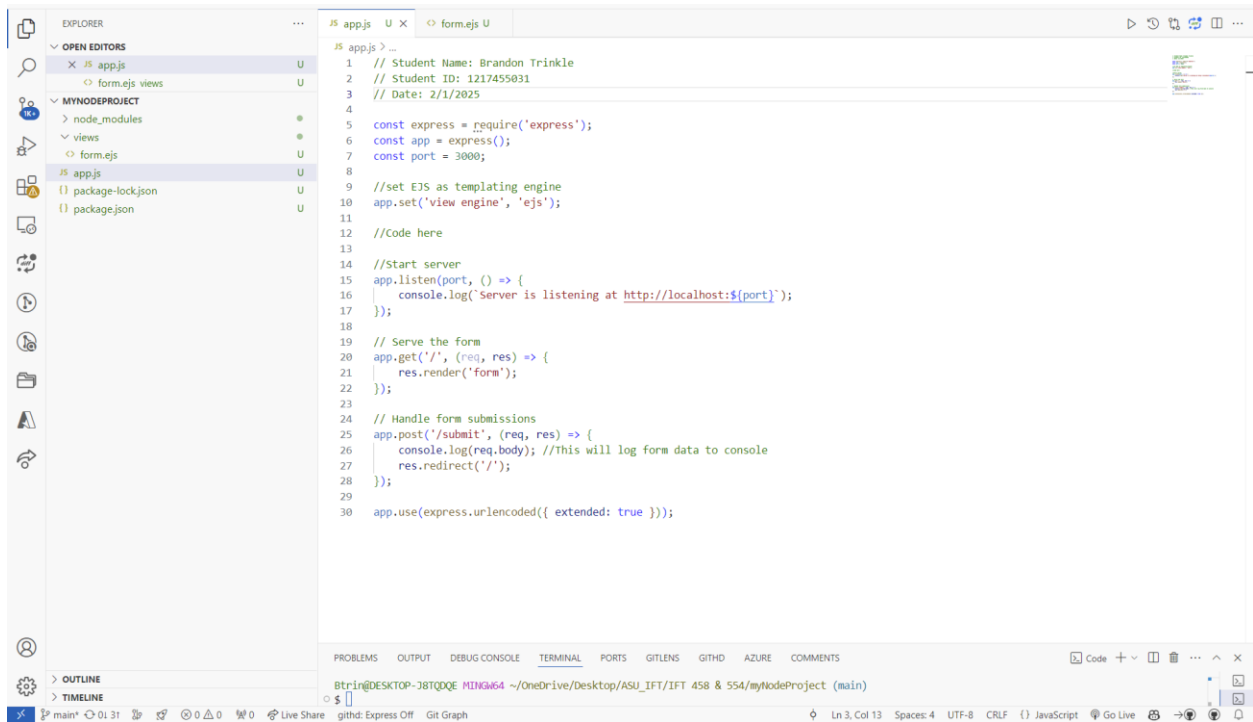
Btr@DESKTOP-J8TQDQE MINGW64 ~/OneDrive/Desktop/ASU_IFT/IFT 458 & 554/myNodeProject (main)
$ npm install ejs
up to date, audited 86 packages in 909ms

16 packages are looking for funding
run 'npm fund' for details

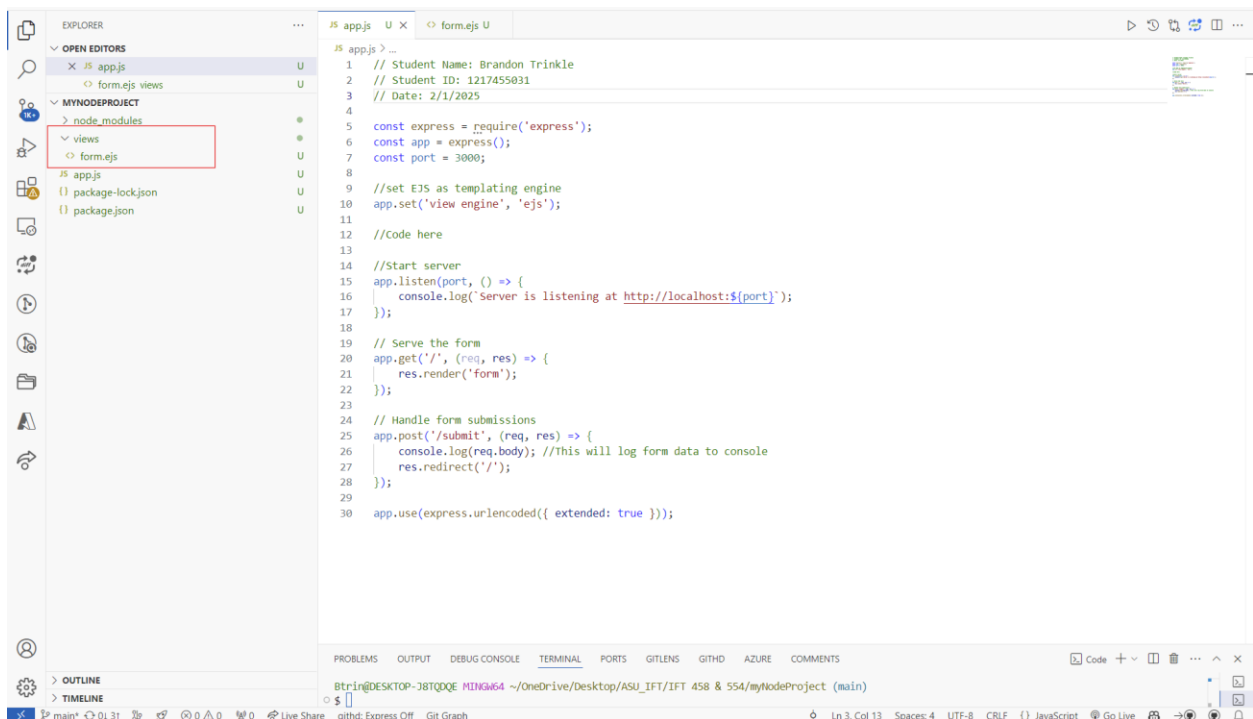
found 0 vulnerabilities

Btr@DESKTOP-J8TQDQE MINGW64 ~/OneDrive/Desktop/ASU_IFT/IFT 458 & 554/myNodeProject (main)
$
  
```

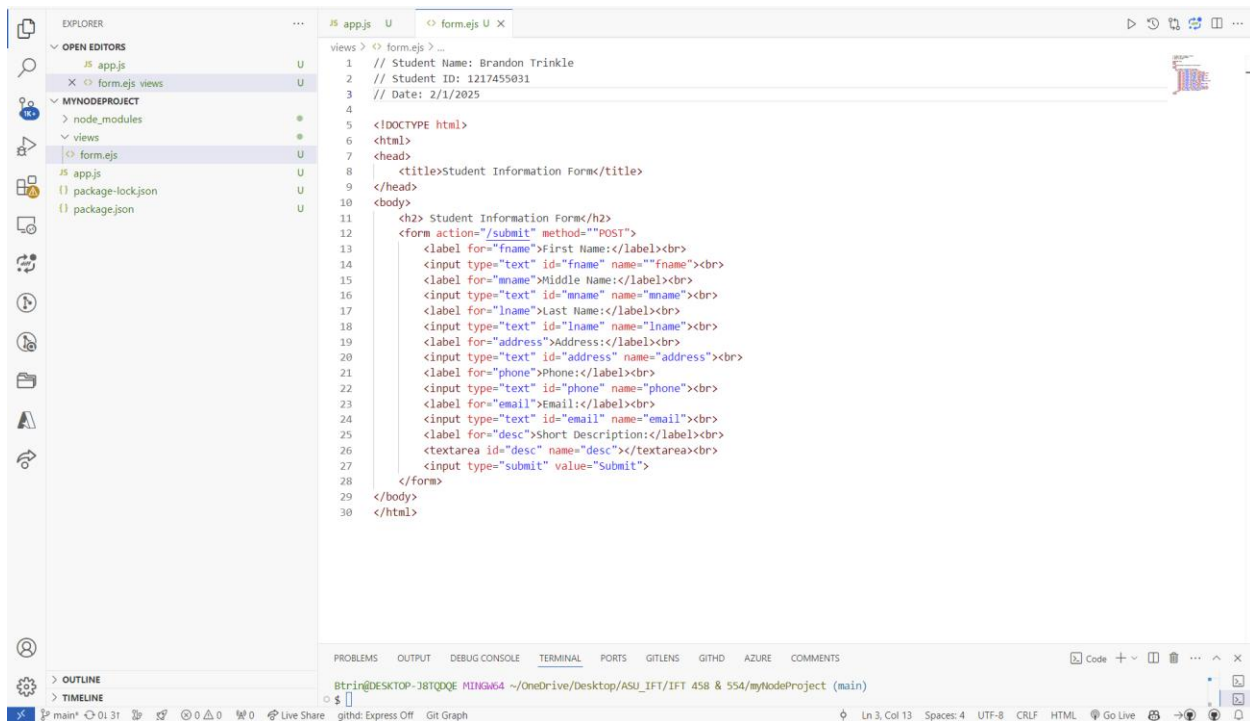
3. `app.js` file open in a text editor, displaying the code for setting up the Express.js server.



4. *Project directory structure in the text editor, showing the newly created views folder.*



5. *form.ejs file open in the text editor, displaying the HTML code for the student information form.*

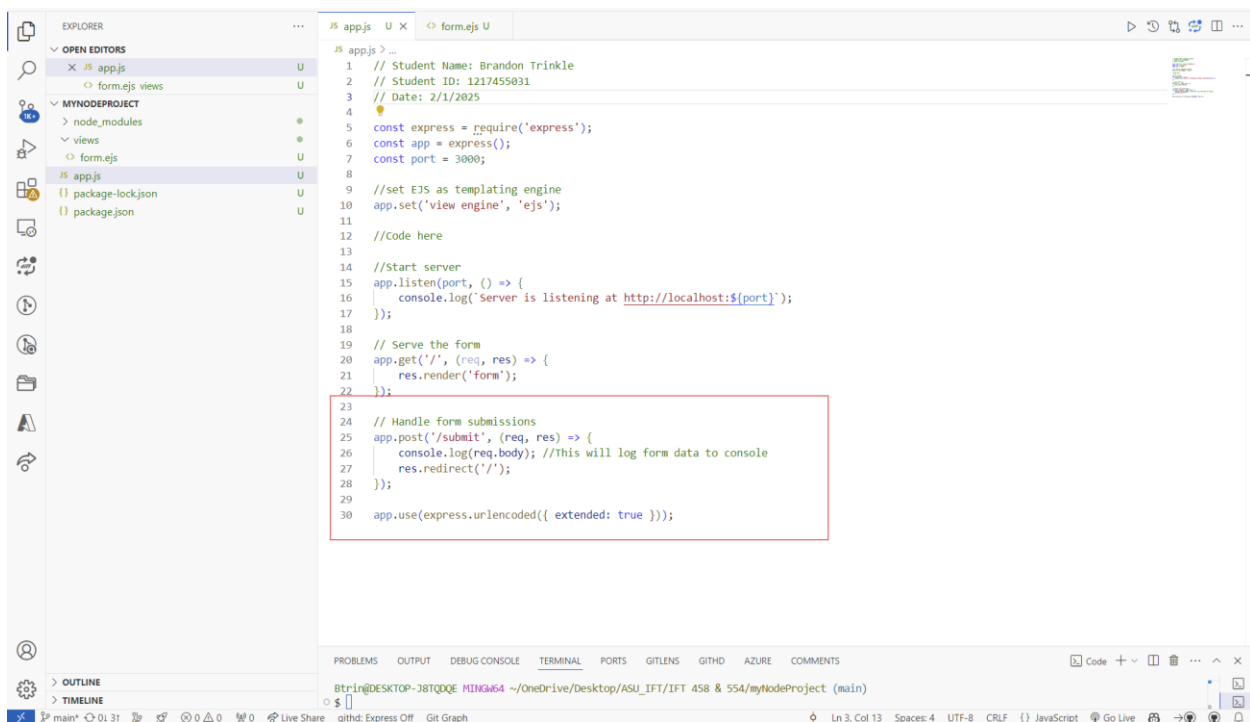


```

1 // Student Name: Brandon Trinkle
2 // Student ID: 1217455031
3 // Date: 2/1/2025
4
5 <!DOCTYPE html>
6 <html>
7 <head>
8   <title>Student Information Form</title>
9 </head>
10 <body>
11   <h2> Student Information Form</h2>
12   <form action="/submit" method="POST">
13     <label for="fname">First Name:</label><br>
14     <input type="text" id="fname" name="fname"><br>
15     <label for="mname">Middle Name:</label><br>
16     <input type="text" id="mname" name="mname"><br>
17     <label for="lname">Last Name:</label><br>
18     <input type="text" id="lname" name="lname"><br>
19     <label for="address">Address:</label><br>
20     <input type="text" id="address" name="address"><br>
21     <label for="phone">Phone:</label><br>
22     <input type="text" id="phone" name="phone"><br>
23     <label for="email">Email:</label><br>
24     <input type="text" id="email" name="email"><br>
25     <label for="desc">Short Description:</label><br>
26     <textarea id="desc" name="desc"></textarea><br>
27     <input type="submit" value="Submit">
28   </form>
29 </body>
30 </html>

```

6. *app.js file showing the added code for serving the form and handling form submissions.*



```

1 // Student Name: Brandon Trinkle
2 // Student ID: 1217455031
3 // Date: 2/1/2025
4
5 const express = require('express');
6 const app = express();
7 const port = 3000;
8
9 //set EJS as templating engine
10 app.set('view engine', 'ejs');
11
12 //Code here
13
14 //Start server
15 app.listen(port, () => {
16   console.log('Server is listening at http://localhost:${port}');
17 });
18
19 // Serve the form
20 app.get('/', (req, res) => {
21   res.render('form');
22 });
23
24 // Handle form submissions
25 app.post('/submit', (req, res) => {
26   console.log(req.body); //this will log form data to console
27   res.redirect('/');
28 });
29
30 app.use(express.urlencoded({ extended: true }));

```

7. *Terminal executing the node app.js command to run the Node.js application.*

```

1 // Student Name: Brandon Trinkle
2 // Student ID: 1217455031
3 // Date: 2/1/2025
4
5 const express = require('express');
6 const app = express();
7 const port = 3000;
8
9 //set EJS as templating engine
10 app.set('view engine', 'ejs');
11
12 //code here
13
14 //Start server
15 app.listen(port, () => {
16   console.log('Server is listening at http://localhost:${port}');
17 });
18
19 // Serve the form
20 app.get('/', (req, res) => {

```

Btrink@DESKTOP-J8TQDQE MINGW64 ~/OneDrive/Desktop/ASU_IFT/IFT 458 & 554/myNodeProject (main)
 \$ node "c:\Users\btrink\OneDrive\Desktop\ASU_IFT\IFT 458 & 554\myNodeProject\app.js"
 Server is listening at http://localhost:3000

8. Web browser displaying the student information form at <http://localhost:3000>.

Module 2 Lab 1: Building a Sim x Student Information Form x +

localhost:3000

App Store | Apporro IP Subnet Calculator Adobe Acrobat All Bookmarks

Student Information Form

First Name:

Middle Name:

Last Name:

Address:

Phone:

Email:

Short Description: