

Assignment 2.1

Project Overview: This project shows how to use Node.js and Express.js to develop a basic web application. The application displays a simple "Hello, World!" greeting in an HTML file. This assignment aims to help students learn how to set up an Express.js server and serve static files for an assignment.

Step 1-

Create your project folder.

Make a folder for your project (Assignment 2.1, for example).

Make a public subfolder inside this folder to house all static files, such as HTML, CSS, and JavaScript.

Step 2-

Install Dependencies

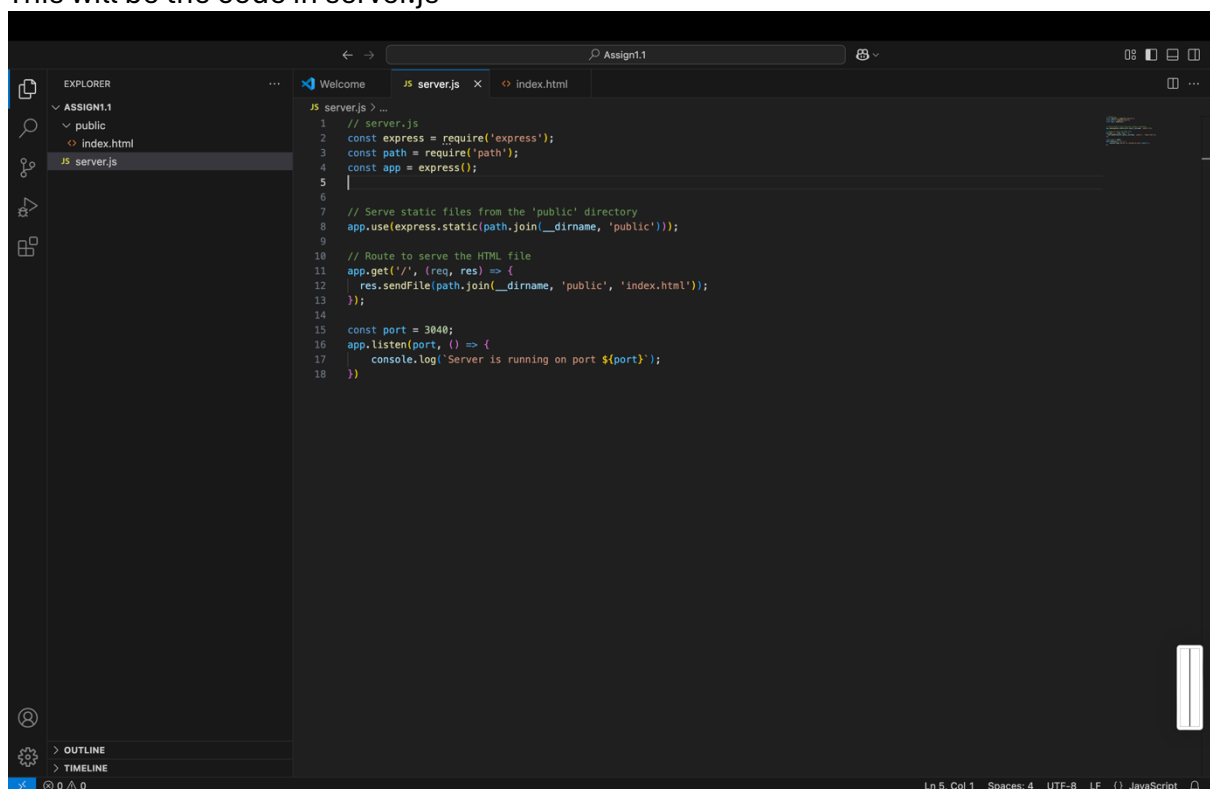
a. Initialize Node.js with the command “npm init -y”

b. Install express with the command “npm install express”

Step 3-

Create a file “server.js” in the root of the project folder.

This will be the code in server.js

A screenshot of a code editor interface, likely Visual Studio Code, showing a project named 'Assign1.1'. The Explorer sidebar on the left shows the file structure: 'ASSIGN1.1' containing a 'public' folder with 'index.html' and a 'server.js' file in the root. The main editor area displays the content of 'server.js'. The code initializes Express.js, serves static files from the 'public' directory, and defines a route for '/'. It also sets a listening port of 3840 and logs the server status. The status bar at the bottom indicates the file is 'Ln 5, Col 1', uses 'UTF-8' encoding, and is a 'JavaScript' file.

```
1 // server.js
2 const express = require('express');
3 const path = require('path');
4 const app = express();
5
6
7 // Serve static files from the 'public' directory
8 app.use(express.static(path.join(__dirname, 'public')));
9
10
11 // Route to serve the HTML file
12 app.get('/', (req, res) => {
13   res.sendFile(path.join(__dirname, 'public', 'index.html'));
14 });
15
16 const port = 3840;
17 app.listen(port, () => {
18   console.log('Server is running on port ${port}');
19 })
```

Explanation-

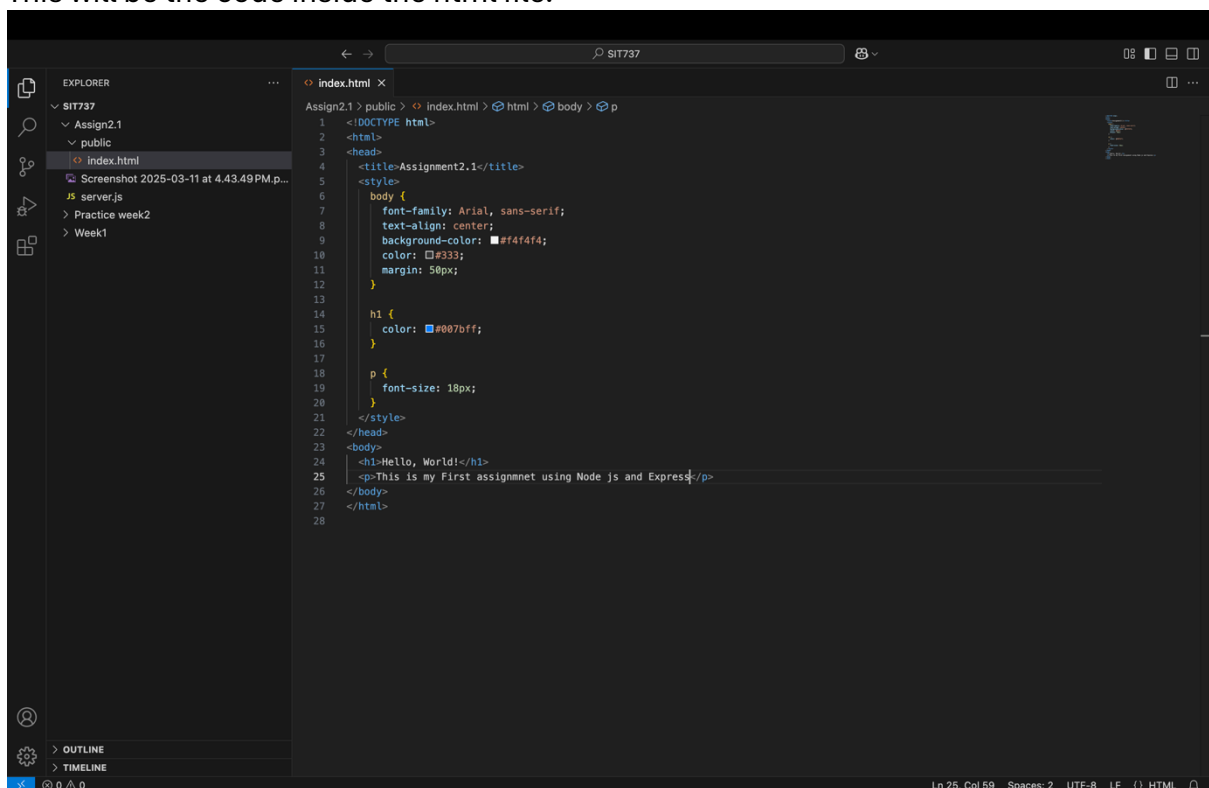
- the server is created by importing Express.

- To guarantee that file paths are accurate across many operating systems, utilize path.
- Static files (HTML, CSS, JS, and pictures) are served from the public folder using the express. static middleware.
- When the root URL is requested, the index.html file is served using the route `app.get('/')`.
- On port 3040, the server is configured to listen.

Step 4-

Create a public folder, and inside this folder, create a file index.html.

This will be the code inside the html file.



The screenshot shows a VS Code editor window with the file explorer on the left and the code editor in the center. The file explorer shows a project structure with folders 'SIT737', 'Assign2.1', 'public', and 'Week1'. The 'public' folder is expanded, showing 'index.html'. The code editor displays the content of 'index.html' with the following code:

```

1  <!DOCTYPE html>
2  <html>
3  <head>
4    <title>Assignment2.1</title>
5  </head>
6  <body>
7    <style>
8      font-family: Arial, sans-serif;
9      text-align: center;
10     background-color: #f4f4f4;
11     color: #333;
12     margin: 50px;
13   </style>
14   <h1>
15     color: #007bff;
16   </h1>
17   <p>
18     font-size: 18px;
19   </p>
20 </body>
21 </html>
22
23 <h1>Hello, World!</h1>
24 <p>This is my First assignment using Node js and Express</p>
25
26 </body>
27 </html>
28

```

Explanation-

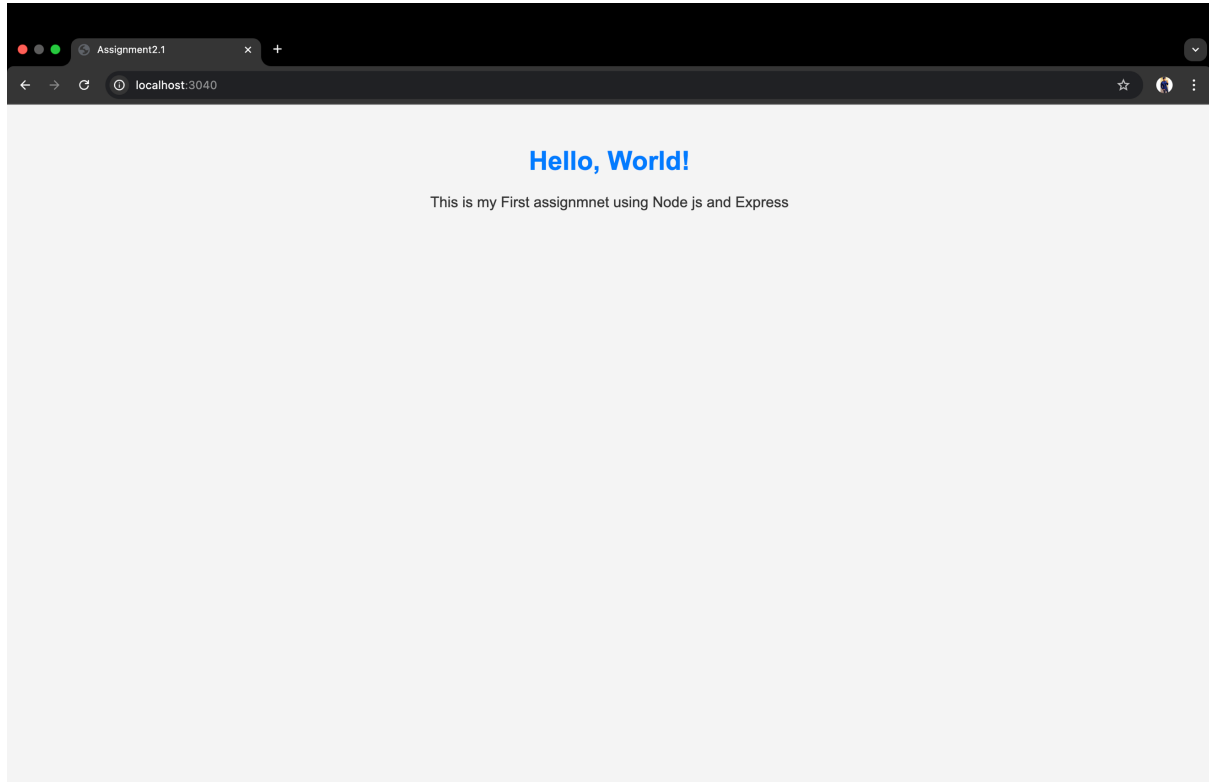
- This HTML document has a paragraph (`<p>`) that shows a message and a simple heading (`<h1>`).
- The `<style>` tag contains the styles that specify the fundamental styles for the paragraph, heading, and body.

Step 5:

- Launch the server
- Open the console and find server.js in the root folder of your project.
- To launch the server, execute the following command:
Node.js server

- Go to <http://localhost:3040> in your browser. The "Hello, World!" message ought should appear.

Output-



GIT HUB LINK-

<https://github.com/Preeyan19/sit737-2025-prac2p.git>