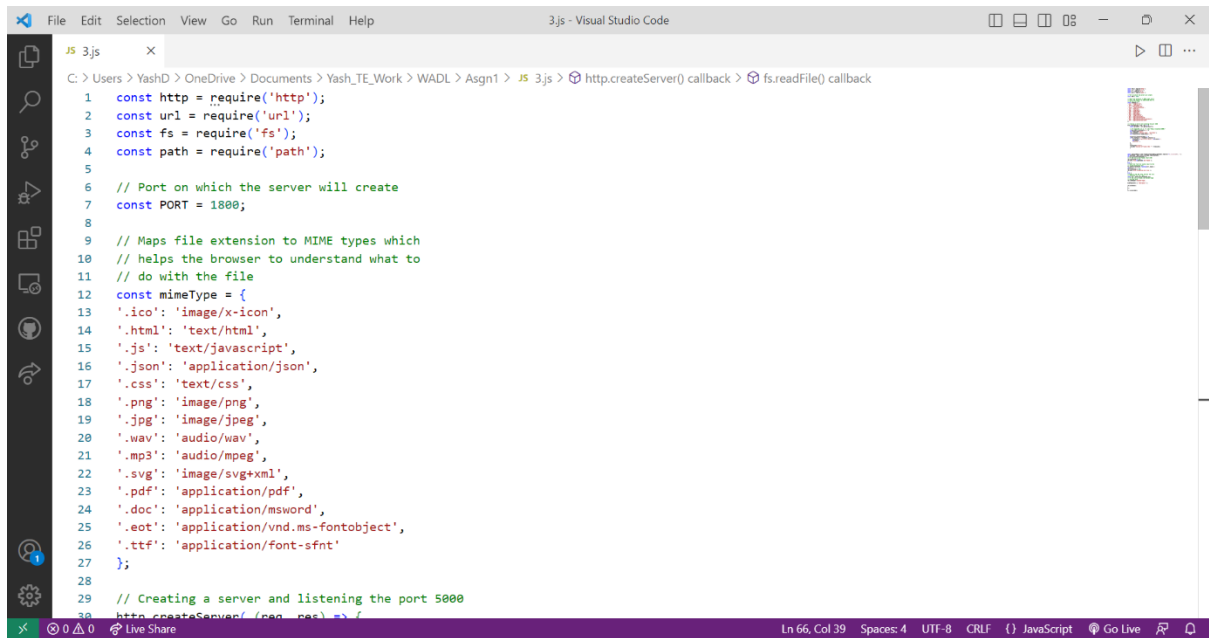


Assignment 3a – 33122 Yash Dhavale

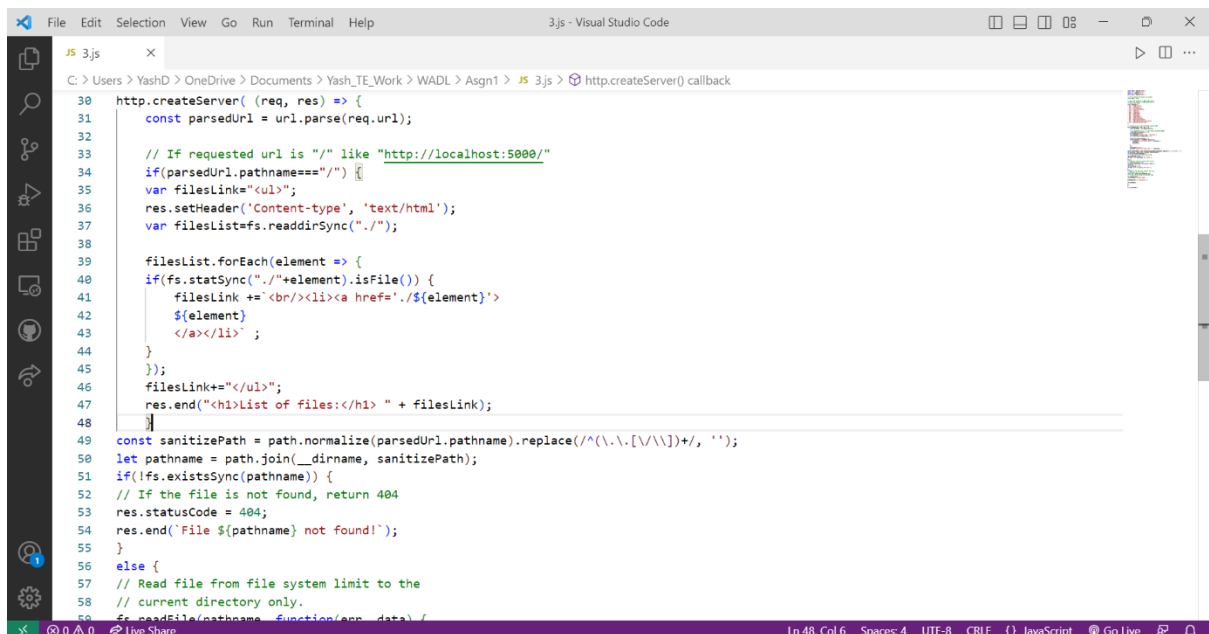
Problem Statement: Create a node.js application to serve a static web site

Output:



```
1 const http = require('http');
2 const url = require('url');
3 const fs = require('fs');
4 const path = require('path');
5
6 // Port on which the server will create
7 const PORT = 1800;
8
9 // Maps file extension to MIME types which
10 // helps the browser to understand what to
11 // do with the file
12 const mimeType = {
13   '.ico': 'image/x-icon',
14   '.html': 'text/html',
15   '.js': 'text/javascript',
16   '.json': 'application/json',
17   '.css': 'text/css',
18   '.png': 'image/png',
19   '.jpg': 'image/jpeg',
20   '.wav': 'audio/wav',
21   '.mp3': 'audio/mpeg',
22   '.svg': 'image/svg+xml',
23   '.pdf': 'application/pdf',
24   '.doc': 'application/msword',
25   '.eot': 'application/vnd.ms-fontobject',
26   '.ttf': 'application/font-sfnt'
27 };
28
29 // Creating a server and listening the port 5000
30 http.createServer((req, res) => {
```

Javascript(Node) code screenshot 1

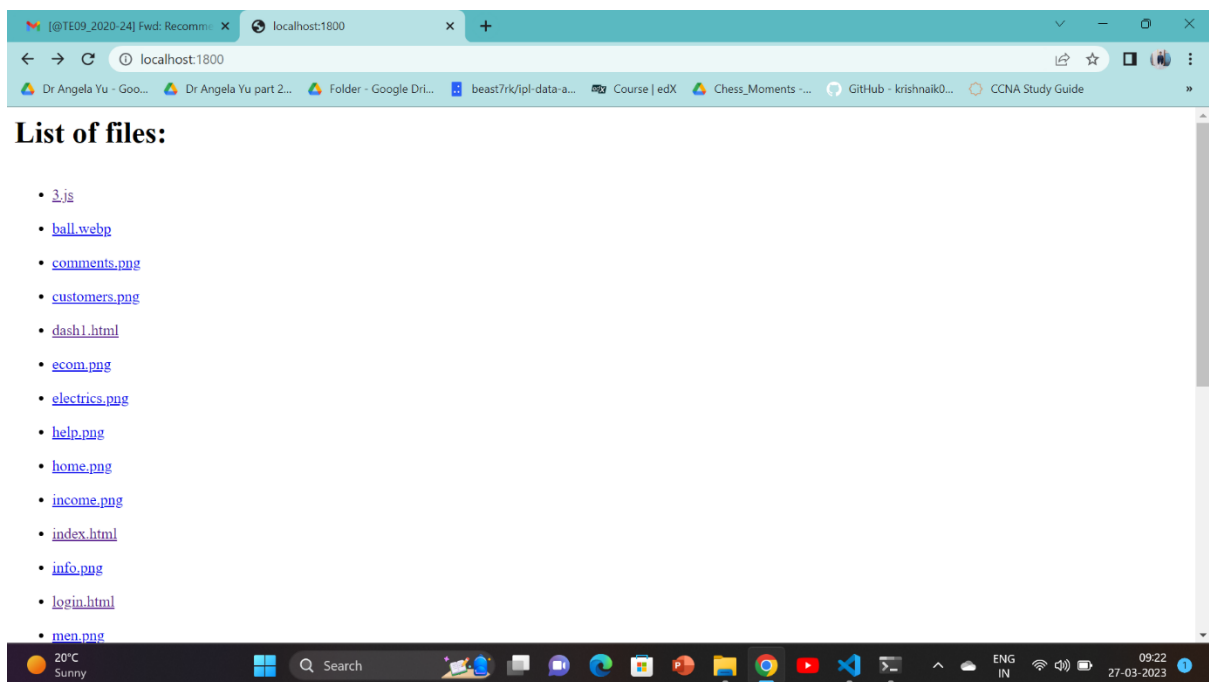


```
30 http.createServer((req, res) => {
31   const parsedUrl = url.parse(req.url);
32
33   // If requested url is "/" like "http://localhost:5000/"
34   if(parsedUrl.pathname === '/') {
35     var filesLink = "<ul>";
36     res.setHeader('Content-type', 'text/html');
37     var fileList = fs.readdirSync("./");
38
39     fileList.forEach(element => {
40       if(fs.statSync("./" + element).isFile()) {
41         filesLink += "<br><li><a href='./" + element + "'>"
42           + element
43           + "</a></li> ";
44       }
45     });
46     filesLink += "</ul>";
47     res.end("<h1>List of files:</h1> " + filesLink);
48   }
49
50   const sanitizePath = path.normalize(parsedUrl.pathname).replace(/^(\/\.\.\/\.\.\/)+/, '');
51   let pathname = path.join(__dirname, sanitizePath);
52   if(!fs.existsSync(pathname)) {
53     // If the file is not found, return 404
54     res.statusCode = 404;
55     res.end('File ${pathname} not found!');
56   }
57   // Read file from file system limit to the
58   // current directory only.
59   fs.readFile(pathname, function(err, data) {
```

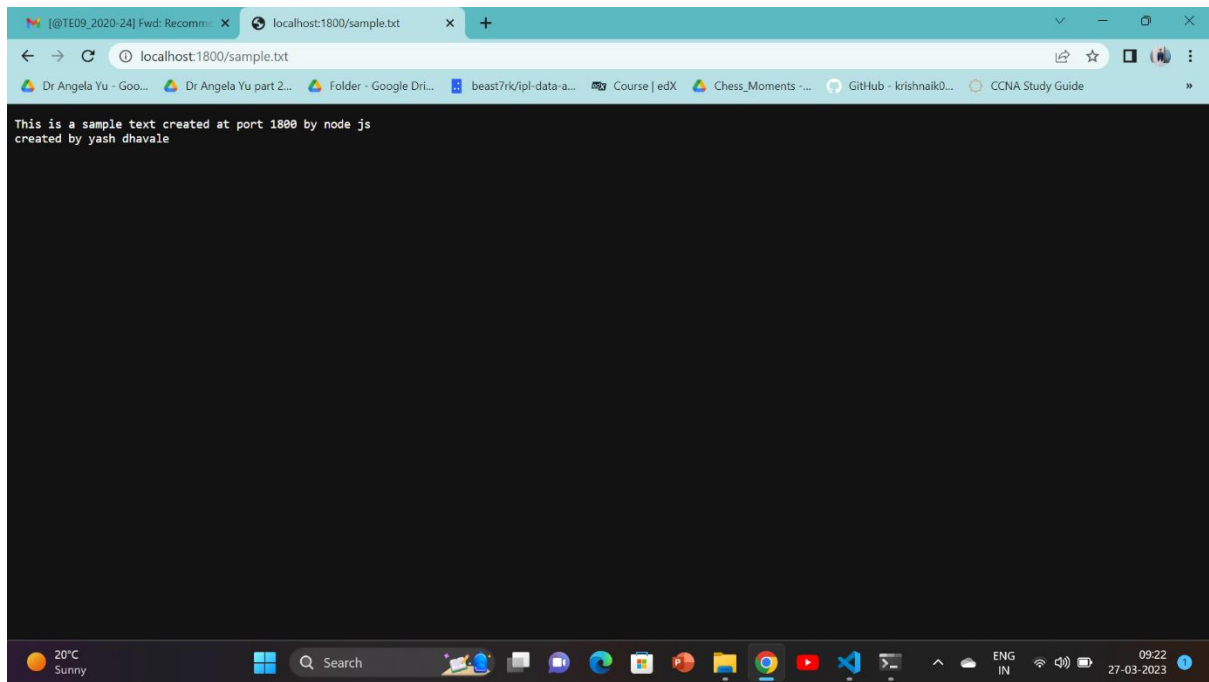
Javascript(Node) code screenshot 2

```
56 else {
57   // Read file from file system limit to the
58   // current directory only.
59   fs.readFile(pathname, function(err, data) {
60     if(err) {
61       res.statusCode = 500;
62       res.end('Error in getting the file.');
```

Javascript(Node) code screenshot 3



Output Screenshot 1 – static web site showing all files



Output Screenshot-2 sample.txt displaying content