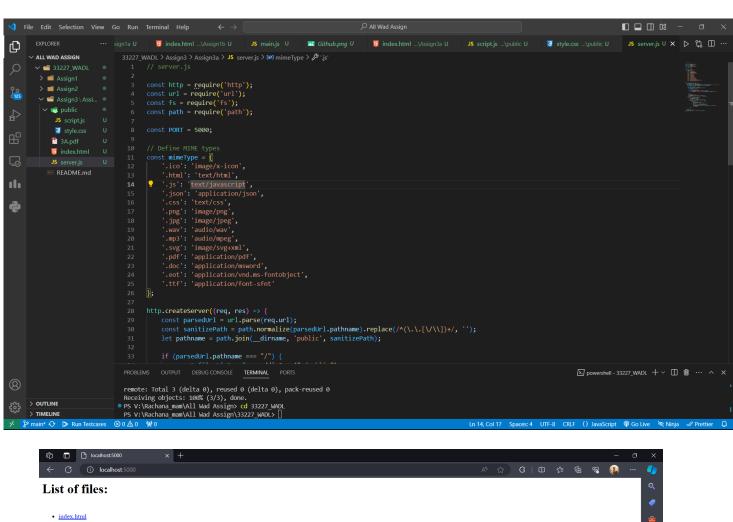
## **Assignment no :- 3A**

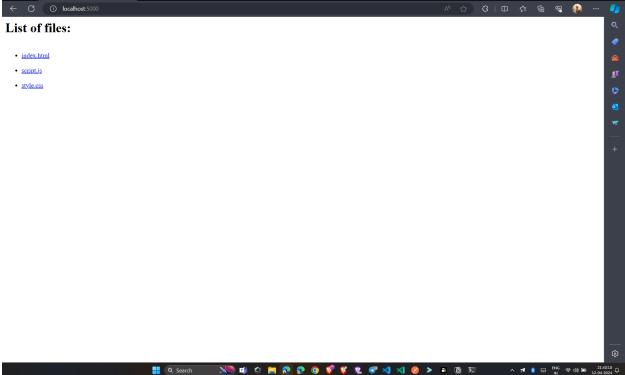
```
// server.js
const http = require('http');
const url = require('url');
const fs = require('fs');
const path = require('path');
const PORT = 5000;
// Define MIME types
const mimeType = {
    '.ico': 'image/x-icon',
    '.html': 'text/html',
    '.js': 'text/javascript',
    '.json': 'application/json',
    '.css': 'text/css',
    '.png': 'image/png',
    '.jpg': 'image/jpeg',
    '.wav': 'audio/wav',
    '.mp3': 'audio/mpeg',
    '.svg': 'image/svg+xml',
    '.pdf': 'application/pdf',
    '.doc': 'application/msword',
    '.eot': 'application/vnd.ms-fontobject',
    '.ttf': 'application/font-sfnt'
};
http.createServer((req, res) => {
    const parsedUrl = url.parse(req.url);
    const sanitizePath = path.normalize(parsedUrl.pathname).rep.
```

Assignment no: - 3A

```
let pathname = path.join(__dirname, 'public', sanitizePath)
   if (parsedUrl.pathname === "/") {
        const filesList = fs.readdirSync("./public");
        let filesLink = "";
        filesList.forEach(element => {
            if (fs.statSync("./public/" + element).isFile()) {
                filesLink += `<br/><a href='./${element}'>$-
        });
        filesLink += "";
        res.setHeader('Content-type', 'text/html');
        res.end("<h1>List of files:</h1> " + filesLink);
   } else {
        if (!fs.existsSync(pathname)) {
            res statusCode = 404;
            res.end(`File ${pathname} not found!`);
       } else {
            fs.readFile(pathname, function (err, data) {
                if (err) {
                    res_statusCode = 500;
                    res.end(`Error in getting the file.`);
                } else {
                    const ext = path.parse(pathname).ext;
                    res.setHeader('Content-type', mimeType[ext]
                    res.end(data);
           });
       }
}).listen(PORT, () => {
   console.log(`Server is running on port ${PORT}`);
});
```

Assignment no:- 3A





Assignment no: - 3A 3