

server/index.js

```
30 const express = require("express");
31 const fs = require("fs");
32 const glob = require("glob");
33 const multer = require("multer");
34
35 const app = express();
36 const port = 3001;
37
38 // Configure Multer for file uploads
39 const upload = multer({ dest: "uploads/" });
40
41 // Set up static file serving for downloads
42 app.use("/downloads", express.static("downloads"));
43
44 // Delete old files on server start
45 const clearFiles = () => {
46   const filelist = glob.sync("uploads/*");
47   filelist.forEach((file) => {
48     fs.unlinkSync(file);
49   });
50 };
51 clearFiles();
52
53 // Route for the home page
54 app.get("/", (req, res) => {
55   res.render("home"); // Replace with the actual home.html content
56 });
57
58 // Route for compressing files
59 app.post("/compress", upload.single("file"), (req, res) => {
60   const up_file = req.file;
61
62   if (up_file) {
63     // Add the logic for compressing the file and saving it to "downloads" folder
64     // Replace the "./c" command with your compression logic
65
66     // Example: Compressing the file
67     const filename = up_file.originalname;
68     const ftype = "-compressed.bin";
69     fs.renameSync(up_file.path, `uploads/${filename}`);
70     // Add compression logic here
71     fs.renameSync(`uploads/${filename}`, `downloads/${filename}${ftype}`);
72
73     res.render("compress", { check: 1 }); // Replace with the actual compress.html
74     content
75   } else {
76     res.render("compress", { check: -1 }); // Replace with the actual
77     compress.html content
78   }
79 });
80
81 // Route for decompressing files
82 app.post("/decompress", upload.single("file"), (req, res) => {
83   const up_file = req.file;
84
85   if (up_file) {
```

```
84 // Add the logic for decompressing the file and saving it to "downloads"
   folder
85 // Replace the "./d" command with your decompression logic
86
87 // Example: Decompressing the file
88 const filename = up_file.originalname;
89 fs.renameSync(up_file.path, `uploads/${filename}`);
90 // Add decompression logic here
91 fs.renameSync(`uploads/${filename}`, `downloads/${filename}`);
92
93 res.render("decompress", { check: 1 }); // Replace with the actual
decompress.html content
94 } else {
95 res.render("decompress", { check: -1 }); // Replace with the actual
decompress.html content
96 }
97 });
98
99 // Route for downloading files
100 app.get("/download", (req, res) => {
101 // Add logic to handle file download from "downloads" folder
102 // For example:
103 const filename = "example"; // Replace with the actual file name to download
104 const ftype = "-compressed.bin"; // Replace with the actual file type
105
106 const filePath = `downloads/${filename}${ftype}`;
107 res.download(filePath);
108 });
109
110 // Start the server
111 app.listen(port, () => {
112 console.log(`Server has started on port ${port}`);
113 });
```