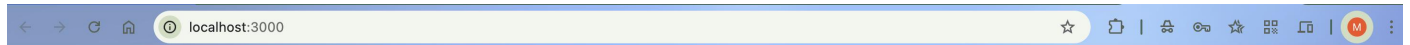


1. Develop nodejs application with following requirements:

- Develop a route `"/gethello"` with GET method. It displays `"Hello NodeJS!!"` as response.
- Make an HTML page and display.
- Call `"/gethello"` route from HTML page using AJAX call. (Any frontend AJAX call API can be used.)

==> Screenshot :



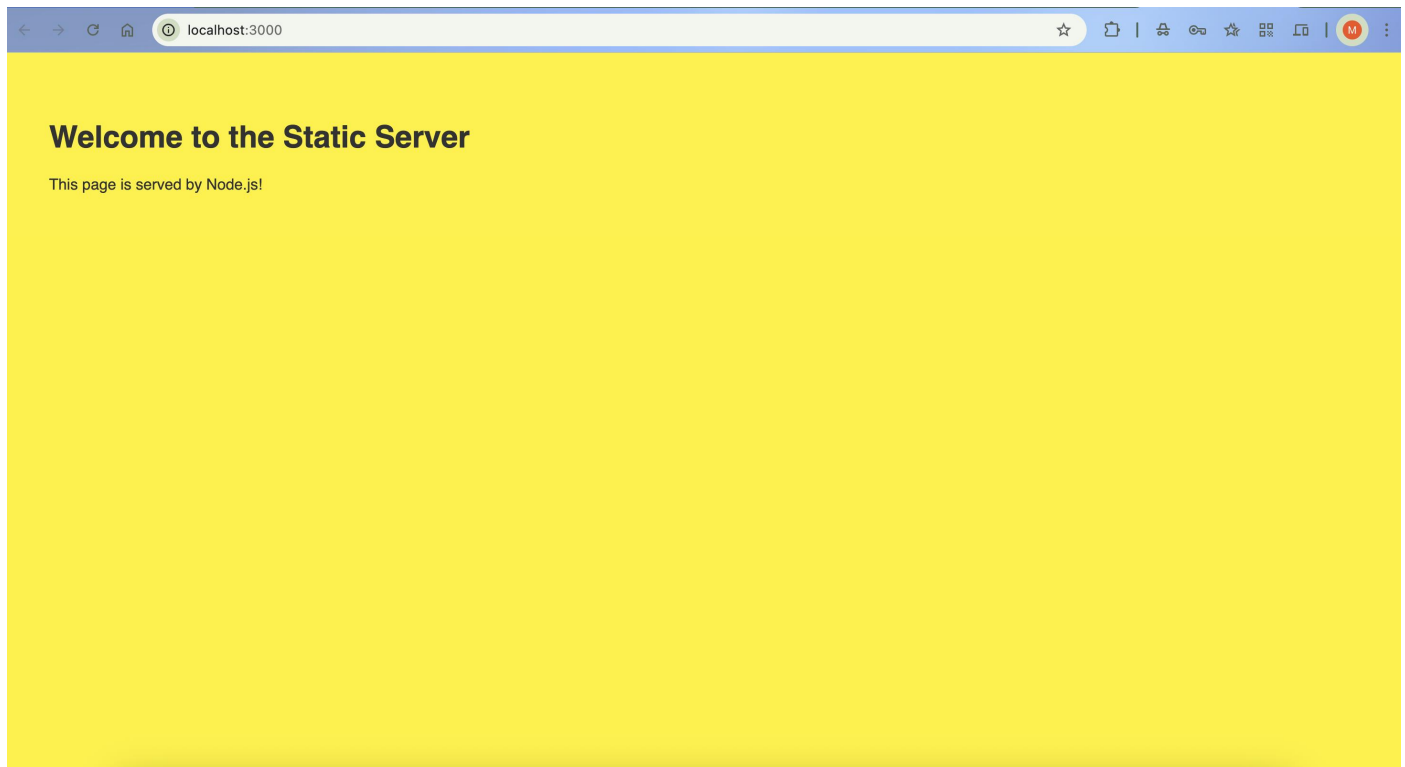
AJAX Call to NodeJS

Call /gethello

Hello NodeJS!!

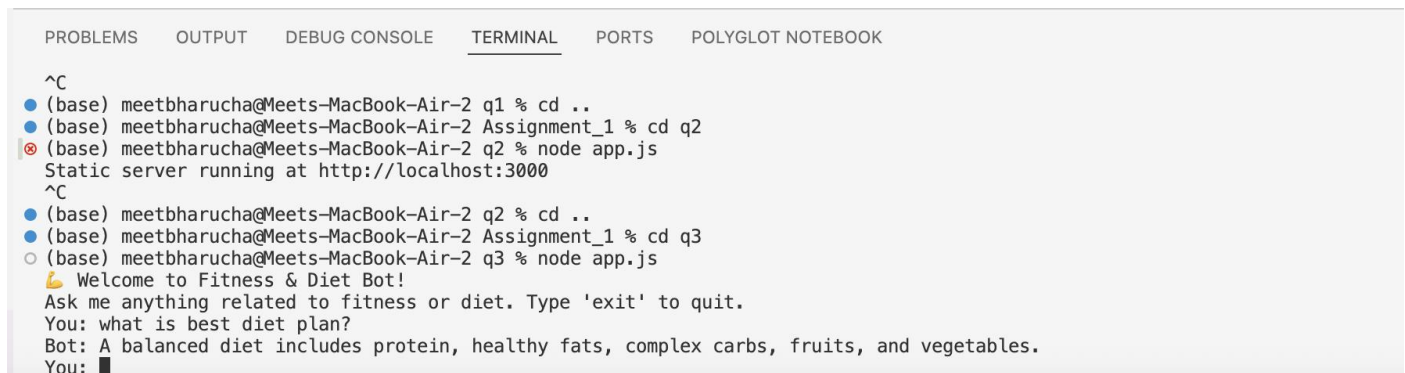
2. Develop a web server which serves static resources.

==> Screenshot :



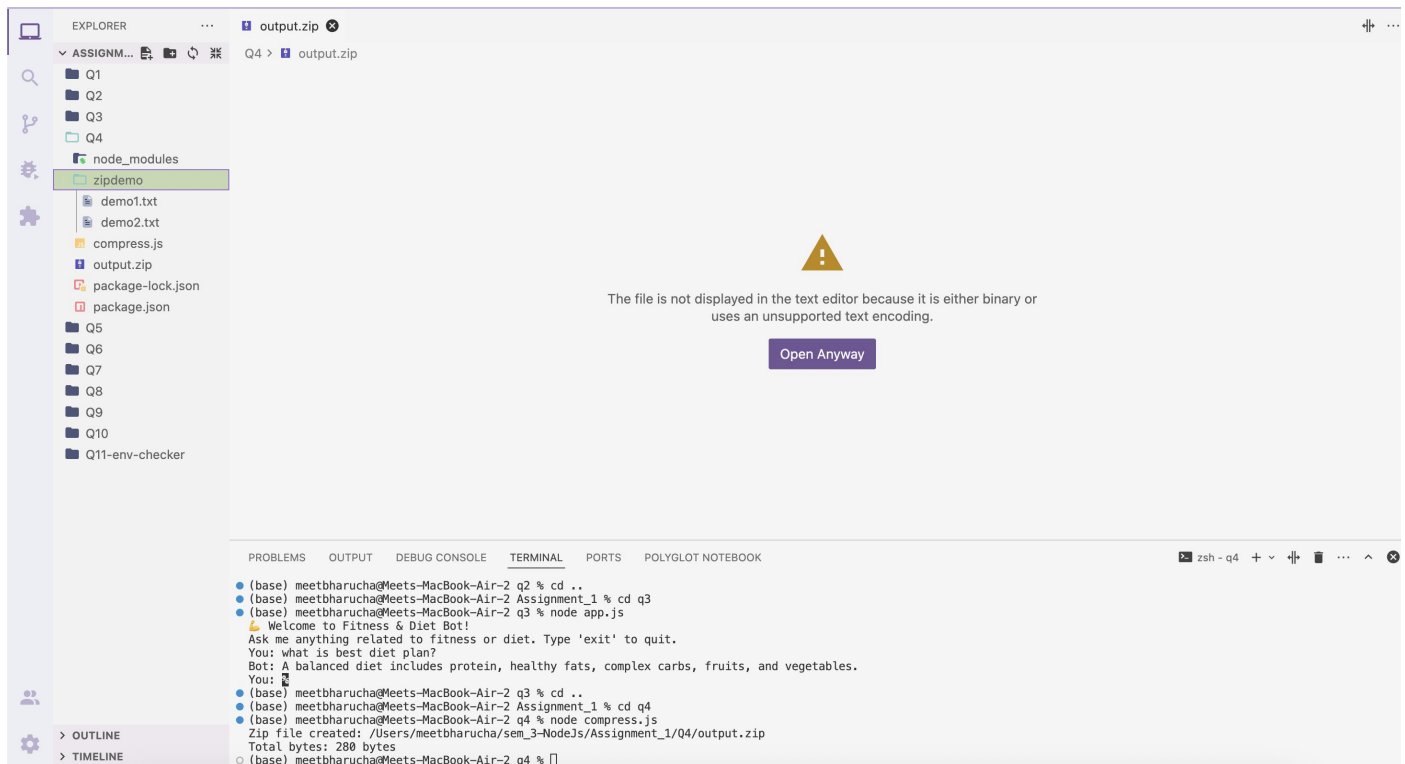
3. Develop a module for domain specific chatbot and use it in a command line application.

==>Screenshot :



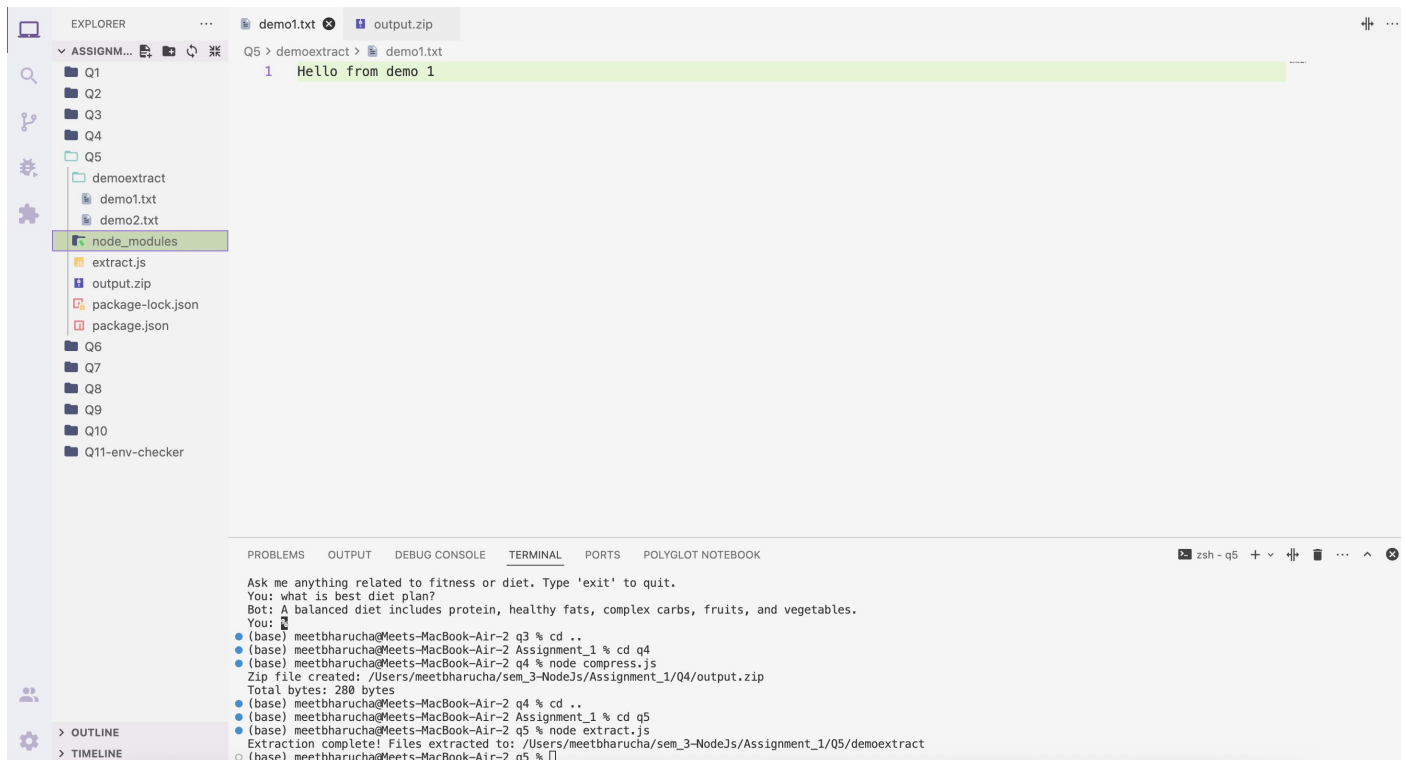
4. Write a program to create a compressed zip file for a folder.

==>Screenshot :



5. Write a program to extract a zip file.

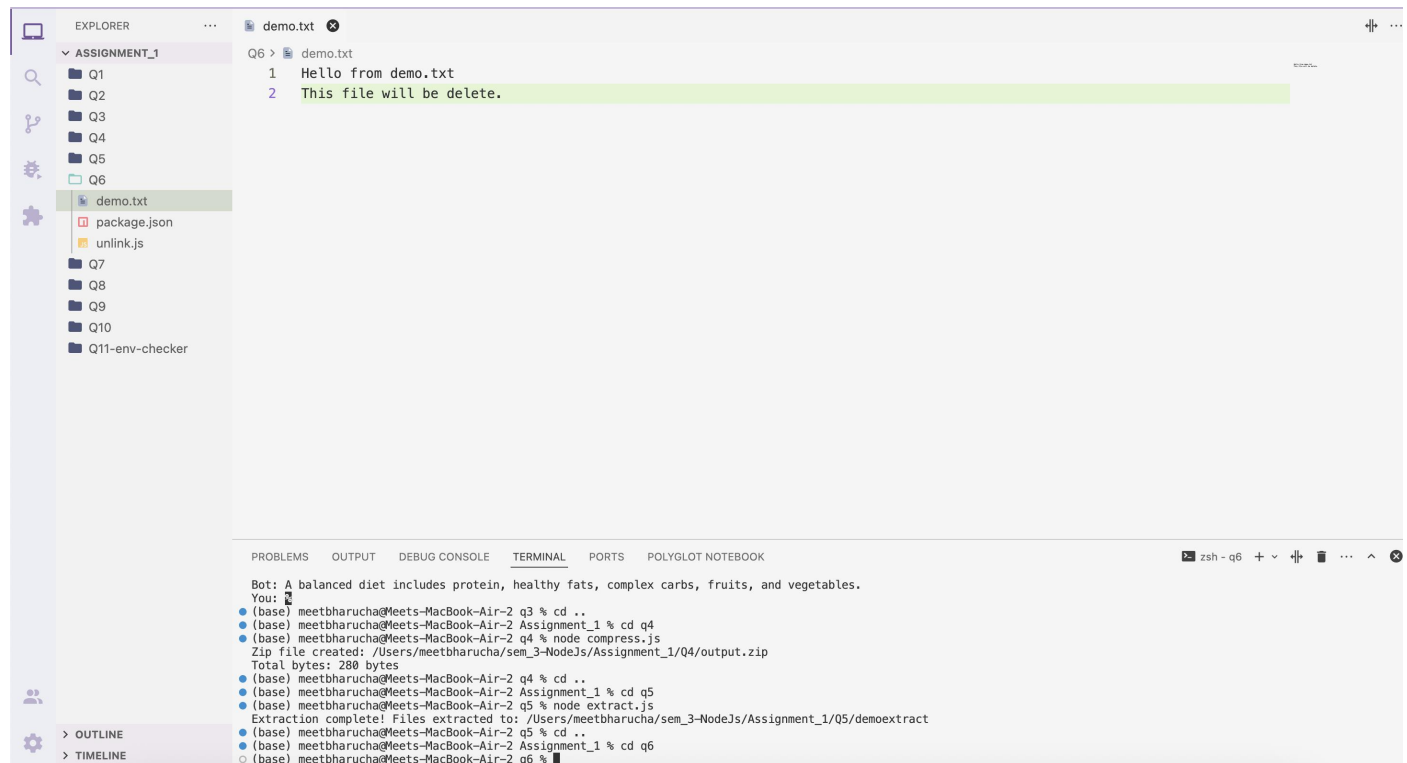
==>Screenshot :



6. Write a program to promisify fs.unlink function and call it.

==>Screenshot :

Before Unlink :

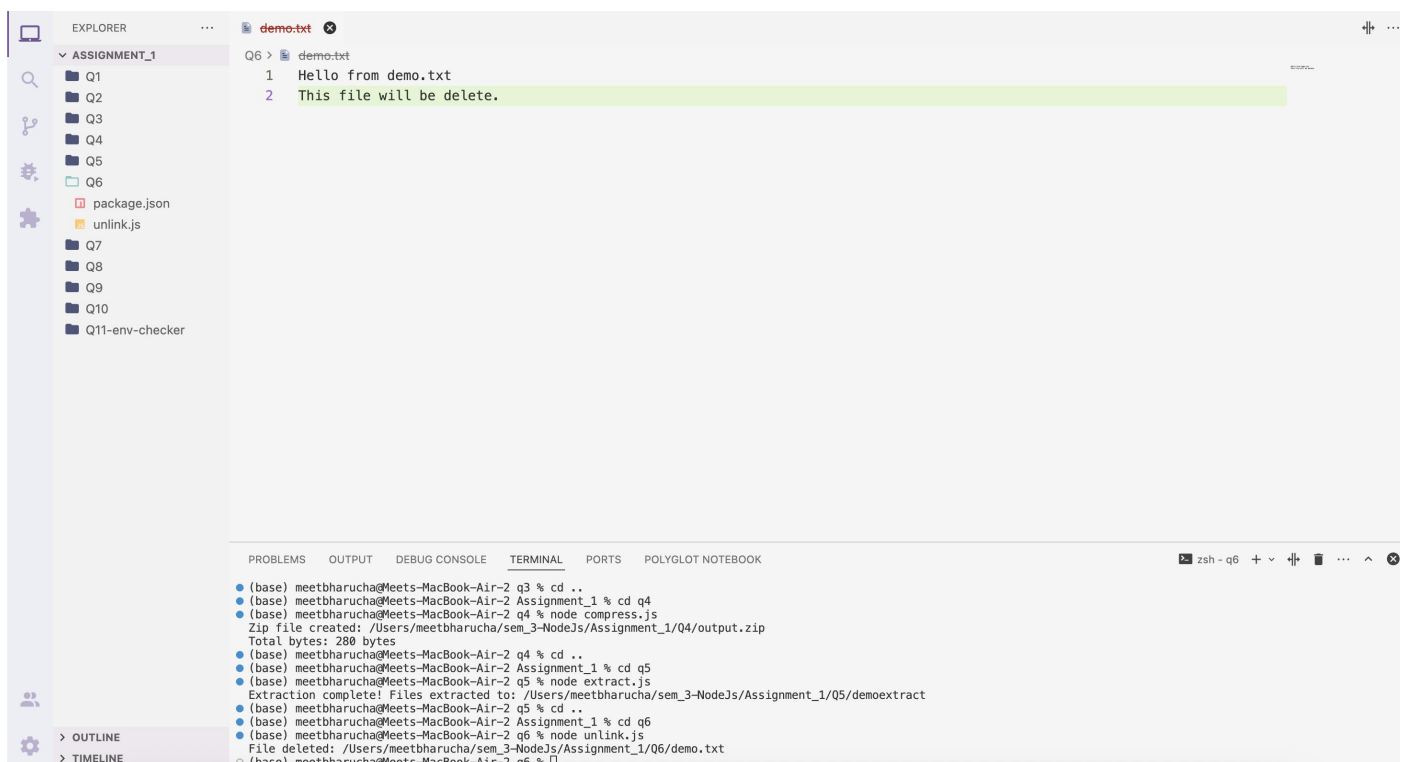


The screenshot shows the Visual Studio Code interface. The Explorer sidebar on the left displays a project structure with a folder named 'ASSIGNMENT_1' containing subfolders Q1 through Q11 and files 'demo.txt', 'package.json', and 'unlink.js'. The main editor window is open to 'demo.txt' in the Q6 folder, showing two lines of text: '1 Hello from demo.txt' and '2 This file will be delete.' The bottom panel shows the TERMINAL tab with a series of shell commands and their outputs, including directory navigation, file compression, and extraction.

```
Q6 > demo.txt
1 Hello from demo.txt
2 This file will be delete.
```

```
Bot: A balanced diet includes protein, healthy fats, complex carbs, fruits, and vegetables.
You:
(base) meetbharucha@Meets-MacBook-Air-2 q3 % cd ..
(base) meetbharucha@Meets-MacBook-Air-2 Assignment_1 % cd q4
(base) meetbharucha@Meets-MacBook-Air-2 q4 % node compress.js
Zip file created: /Users/meetbharucha/sem_3-NodeJs/Assignment_1/Q4/output.zip
Total bytes: 280 bytes
(base) meetbharucha@Meets-MacBook-Air-2 q4 % cd ..
(base) meetbharucha@Meets-MacBook-Air-2 Assignment_1 % cd q5
(base) meetbharucha@Meets-MacBook-Air-2 q5 % node extract.js
Extraction complete! Files extracted to: /Users/meetbharucha/sem_3-NodeJs/Assignment_1/Q5/demoeextract
(base) meetbharucha@Meets-MacBook-Air-2 q5 % cd ..
(base) meetbharucha@Meets-MacBook-Air-2 Assignment_1 % cd q6
(base) meetbharucha@Meets-MacBook-Air-2 q6 %
```

After Unlink :



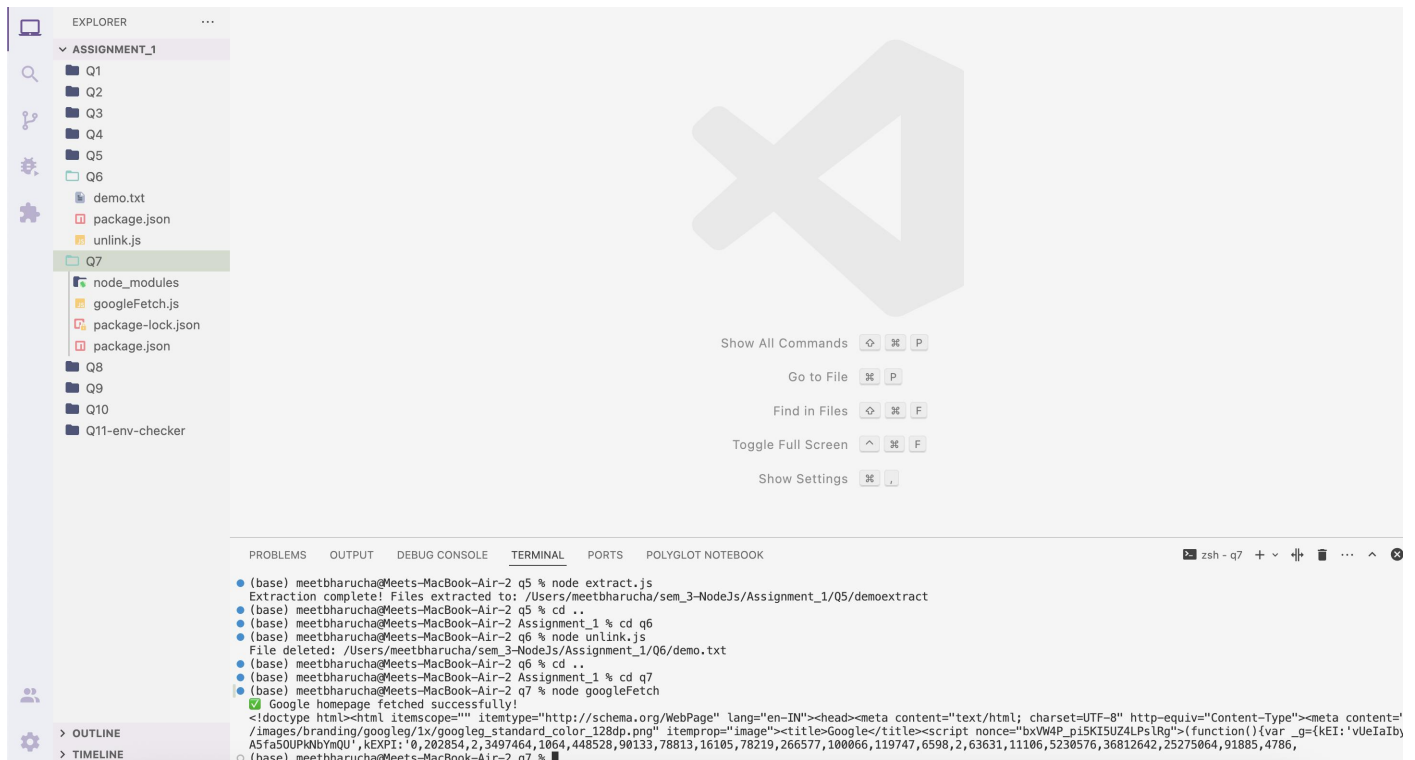
This screenshot is identical to the previous one, showing the 'Before Unlink' state. The 'demo.txt' file still contains the same two lines of text, and the terminal shows the same sequence of commands and outputs.

```
Q6 > demo.txt
1 Hello from demo.txt
2 This file will be delete.
```

```
(base) meetbharucha@Meets-MacBook-Air-2 q3 % cd ..
(base) meetbharucha@Meets-MacBook-Air-2 Assignment_1 % cd q4
(base) meetbharucha@Meets-MacBook-Air-2 q4 % node compress.js
Zip file created: /Users/meetbharucha/sem_3-NodeJs/Assignment_1/Q4/output.zip
Total bytes: 280 bytes
(base) meetbharucha@Meets-MacBook-Air-2 q4 % cd ..
(base) meetbharucha@Meets-MacBook-Air-2 Assignment_1 % cd q5
(base) meetbharucha@Meets-MacBook-Air-2 q5 % node extract.js
Extraction complete! Files extracted to: /Users/meetbharucha/sem_3-NodeJs/Assignment_1/Q5/demoeextract
(base) meetbharucha@Meets-MacBook-Air-2 q5 % cd ..
(base) meetbharucha@Meets-MacBook-Air-2 Assignment_1 % cd q6
(base) meetbharucha@Meets-MacBook-Air-2 q6 %
```

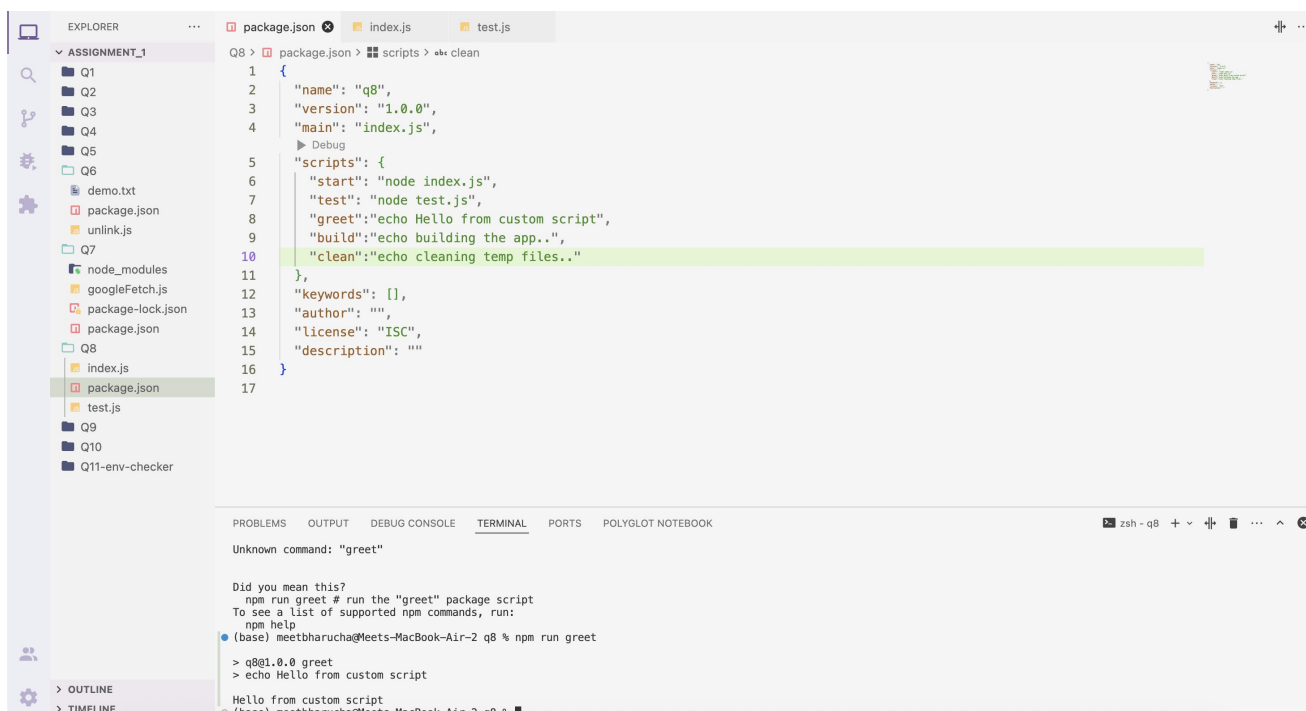
7. Fetch data of google page using note-fetch using async-await model.

==>Screenshot :



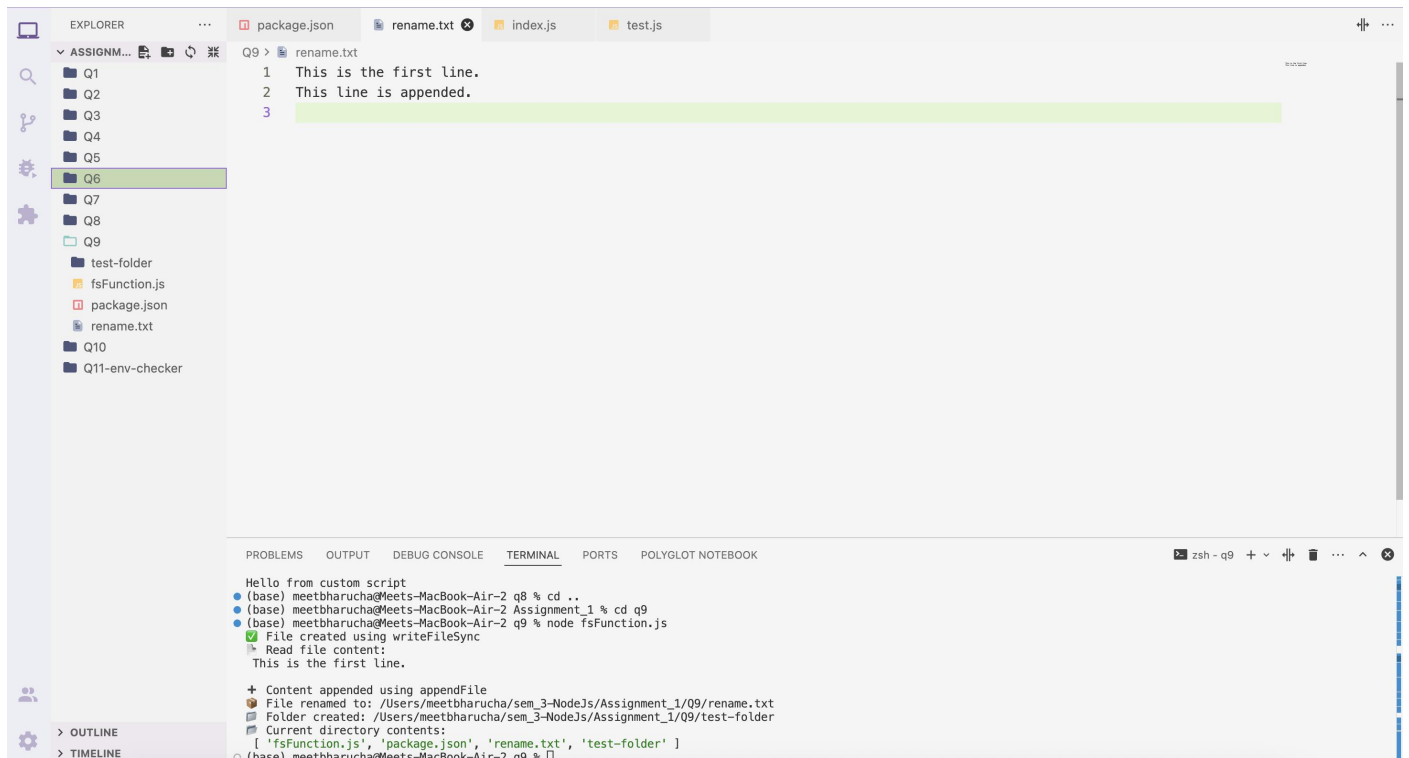
8. Set a server script, a test script and 3 user defined scripts in package.json file in your nodejs application.

==>Screenshot :



9. A program which calls useful functions in fs module.

==>Screenshot :



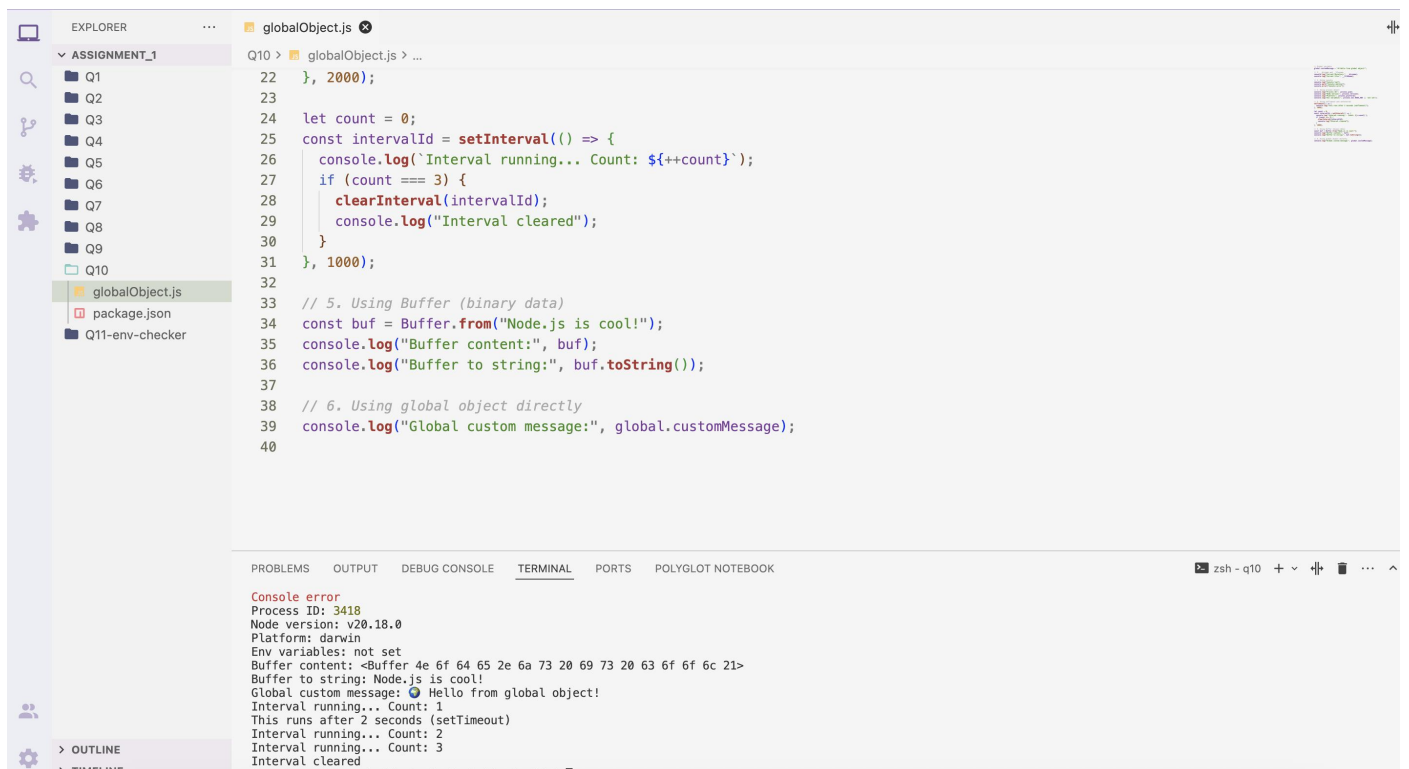
The screenshot shows a VS Code editor with a file explorer on the left and a terminal window at the bottom. The file explorer shows a project structure with folders Q1 through Q11 and files like package.json, rename.txt, index.js, and test.js. The terminal window shows the execution of a Node.js script using the fs module. The script creates a file, appends content, and renames a file. The output of the script is displayed in the terminal.

```
Q9 > rename.txt
1 This is the first line.
2 This line is appended.
3
```

```
zsh - q9
Hello from custom script
(base) meetbharucha@Meets-MacBook-Air-2 q8 % cd ..
(base) meetbharucha@Meets-MacBook-Air-2 Assignment_1 % cd q9
(base) meetbharucha@Meets-MacBook-Air-2 q9 % node fsFunction.js
File created using writeFileSync
Read file content:
This is the first line.
Content appended using appendFile
File renamed to: /Users/meetbharucha/sem_3-NodeJs/Assignment_1/q9/rename.txt
Folder created: /Users/meetbharucha/sem_3-NodeJs/Assignment_1/q9/test-folder
Current directory contents:
[ 'fsFunction.js', 'package.json', 'rename.txt', 'test-folder' ]
(base) meetbharucha@Meets-MacBook-Air-2 q9 %
```

10. A program which uses global objects in nodejs.

==>Screenshot :



The screenshot shows a VS Code editor with a file explorer on the left and a terminal window at the bottom. The file explorer shows a project structure with folders Q1 through Q10 and files like globalObject.js, package.json, and Q11-env-checker. The terminal window shows the execution of a Node.js script using global objects. The script uses setInterval to log a message, Buffer.from to create a buffer, and global.customMessage to log a message. The output of the script is displayed in the terminal.

```
Q10 > globalObject.js > ...
22 }, 2000);
23
24 let count = 0;
25 const intervalId = setInterval(() => {
26   console.log(`Interval running... Count: ${++count}`);
27   if (count === 3) {
28     clearInterval(intervalId);
29     console.log("Interval cleared");
30   }
31 }, 1000);
32
33 // 5. Using Buffer (binary data)
34 const buf = Buffer.from("Node.js is cool!");
35 console.log("Buffer content:", buf);
36 console.log("Buffer to string:", buf.toString());
37
38 // 6. Using global object directly
39 console.log("Global custom message:", global.customMessage);
40
```

```
zsh - q10
Console error
Process ID: 3418
Node version: v20.18.0
Platform: darwin
Env variables: not set
Buffer content: <Buffer 4e 6f 64 65 2e 6a 73 20 69 73 20 63 6f 6f 6c 21>
Buffer to string: Node.js is cool!
Global custom message: Hello from global object!
Interval running... Count: 1
This runs after 2 seconds (setTimeout)
Interval running... Count: 2
Interval running... Count: 3
Interval cleared
(base) meetbharucha@Meets-MacBook-Air-2 q10 %
```

11. Develop a useful package and publish it on npmjs.com

==>Screenshot :

The screenshot shows the npmjs.com website with the user settings page for 'meet_bharucha'. The browser address bar shows 'npmjs.com/settings/meet_bharucha/packages'. The top navigation bar includes links for 'Pro', 'Teams', 'Pricing', and 'Documentation'. The left sidebar contains a menu with 'Profile', 'Packages', 'Account', 'Billing Info', 'Access Tokens', and 'Organizations' (with a '+'). The main content area is titled '1 packages' and lists a package named 'q11-env-checker'. The package description is 'Validate environment variables with types and defaults before app startup.' and it was published by 'meet_bharucha' as version '1.0.1' 'a minute ago'. The footer contains four columns of links: 'Support' (Help, Advisories, Status), 'Company' (About, Blog, Press), 'Terms & Policies' (Policies, Terms of Use, Code of Conduct), and social media icons for GitHub and npm.

npmjs.com/settings/meet_bharucha/packages

Pro Teams Pricing Documentation

npm Search packages Search

Profile Packages Account Billing Info Access Tokens Organizations +

None

1 packages

q11-env-checker

✓ Validate environment variables with types and defaults before app startup.

meet_bharucha published 1.0.1 • a minute ago

Support: Help, Advisories, Status

Company: About, Blog, Press

Terms & Policies: Policies, Terms of Use, Code of Conduct