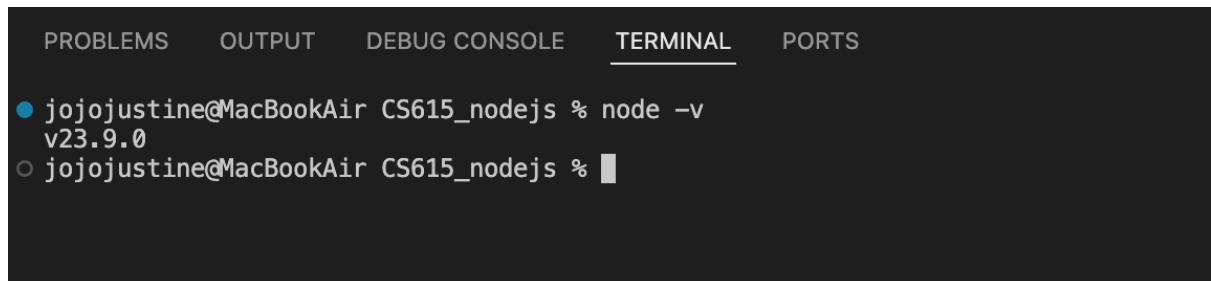


Part 1

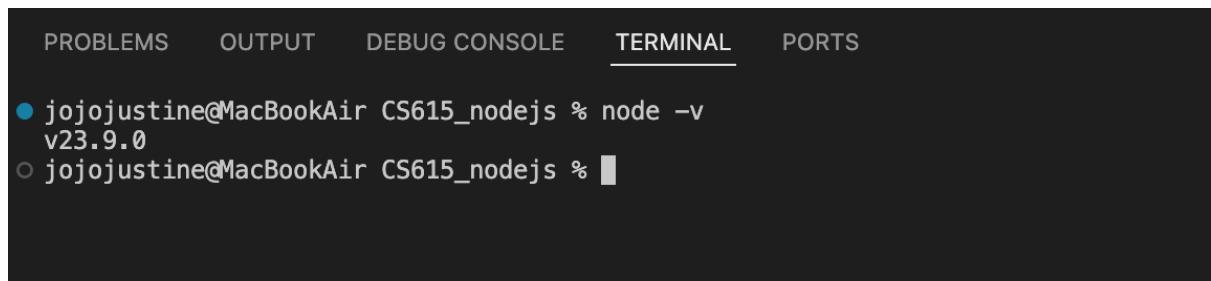
Node Installed



A screenshot of a terminal window with tabs at the top: PROBLEMS, OUTPUT, DEBUG CONSOLE, TERMINAL (underlined), and PORTS. The terminal shows two entries: a blue dot followed by the user's name and host, and an empty circle followed by the same information. The first entry is a command: node -v, which outputs v23.9.0. The second entry is a blank line.

```
PROBLEMS    OUTPUT    DEBUG CONSOLE    TERMINAL    PORTS
● jojojustine@MacBookAir CS615_nodejs % node -v
v23.9.0
○ jojojustine@MacBookAir CS615_nodejs %
```

Npm Installed



A screenshot of a terminal window with tabs at the top: PROBLEMS, OUTPUT, DEBUG CONSOLE, TERMINAL (underlined), and PORTS. The terminal shows two entries: a blue dot followed by the user's name and host, and an empty circle followed by the same information. The first entry is a command: node -v, which outputs v23.9.0. The second entry is a blank line.

```
PROBLEMS    OUTPUT    DEBUG CONSOLE    TERMINAL    PORTS
● jojojustine@MacBookAir CS615_nodejs % node -v
v23.9.0
○ jojojustine@MacBookAir CS615_nodejs %
```

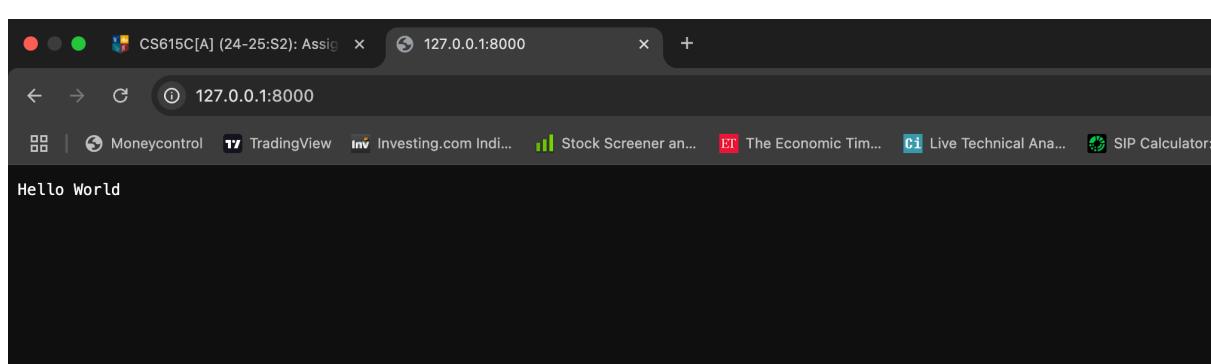
App.js

The screenshot shows the Visual Studio Code interface. On the left is the Explorer sidebar with a project named 'CS615_NODEJS' containing an 'app.js' file. The main area shows the 'app.js' code:

```
1 console.log("Hello World!");
```

. Below the code editor is the Terminal tab, which displays the command `node app.js` and its output: `Hello World!`. The status bar at the bottom indicates the Node.js version is v23.9.0.

```
jojojustine@MacBookAir test-node % node hello.js
Server running at http://127.0.0.1:8000/
```



The screenshot shows the Visual Studio Code (VS Code) interface with a dark theme. The left sidebar contains a file tree with a folder named 'CS615_NODEJS' containing files 'test-node', '1_basic.js', and 'app.js'. The main editor area displays the contents of '1_basic.js':

```
1 setTimeout(function(){
2   console.log('3 seconds have passed');
3 }, 3000);
```

The bottom right corner shows the terminal window with the following output:

```
Node.js v23.9.0
● jojojustine@MacBookAir test-node % cd ..
● jojojustine@MacBookAir CS615_nodejs % node 1_basic.js
  3 seconds have passed
○ jojojustine@MacBookAir CS615_nodejs %
```

The screenshot shows the Visual Studio Code (VS Code) interface with a dark theme. The left sidebar contains icons for Explorer, Search, Open, and Terminal. The Explorer panel shows a folder named 'CS615_NODEJS' containing files: 'test-node', '1_basic.js', and 'app.js'. The main editor area displays the contents of '1_basic.js':

```
1  setTimeout(function(){
2    console.log('3 seconds have passed');
3  }, 3000);
4  var time = 0;
5  var timer = setInterval(function(){
6    time += 2;
7    console.log(time + ' seconds have passed!')
8    //if (time > 5){ // add this code later
9    // clearInterval(timer); // (i.e. run code first without the lines
10   // } // in blue font to see what happens
11 },
12 2000);
```

The bottom right corner shows the terminal window with the following output:

```
jojojustine@MacBookAir CS615_nodejs % node 1_basic.js
3 seconds have passed
2 seconds have passed!
3 seconds have passed
4 seconds have passed!
5 seconds have passed!
```

The screenshot shows the Visual Studio Code interface with a dark theme. The Explorer sidebar on the left lists a folder named 'CS615_NODEJS' containing files: 'test-node', '1_basic.js', and 'app.js'. The '1_basic.js' file is open in the main editor area, displaying the following code:

```
1 setTimeout(function(){
2     console.log('3 seconds have passed!');
3 }, 3000);
4 var time = 0;
5 var timer = setInterval(function(){
6     time += 2;
7     console.log(time + ' seconds have passed!')
8     if (time > 5){ // add this code later
9         clearInterval(timer); // (i.e. run code first without the lines
10        } // in blue font to see what happens)
11    },
12    2000);
```

The terminal at the bottom shows the output of running the script:

```
^C
jojojustine@MacBookAir CS615_nodejs % node 1_basic.js
2 seconds have passed!
3 seconds have passed!
4 seconds have passed!
5 seconds have passed!
```

The status bar at the bottom right indicates 'Ln 10, Col 4'.

The screenshot shows the VS Code interface with the following details:

- EXPLORER** sidebar: Shows a folder named "CS615_NODEJS" containing files: "test-node", "1_basic.js", and "app.js".
- CODE EDITOR**: The active tab is "1_basic.js" which contains the following code:

```
1 settimeout(function(){
2     console.log('3 seconds have passed');
3     }, 3000);
4     console.log(__dirname);
5     console.log(__filename);
6     var time = 0;
7     var timer = setInterval(function(){
8         time += 2;
9         console.log(time + ' seconds have passed!');
10        if (time > 5){ // add this code later
11            clearInterval(timer); // (i.e. run code first without the lines
12        } // in blue font to see what happens
13    },
14    2000);
```
- TERMINAL**: The terminal tab is active and shows the output of running the script:

```
16 seconds have passed!
18 seconds have passed!
20 seconds have passed!
~
● jojojustine@MacBookAir CS615_nodejs % node 1_basic.js
2 seconds have passed!
3 seconds have passed
4 seconds have passed!
5 seconds have passed!
6 seconds have passed!
● jojojustine@MacBookAir CS615_nodejs % node 1_basic.js
/Users/jojojustine/Downloads/Study/Semester2/Internetsolutions/git/CS615C/Assignment5/CS615_nodejs
/Users/jojojustine/Downloads/Study/Semester2/Internetsolutions/git/CS615C/Assignment5/CS615_nodejs/1_basic.js
2 seconds have passed!
3 seconds have passed
4 seconds have passed!
```

The screenshot shows the VS Code interface with the following details:

- EXPLORER** sidebar: Shows a folder named "CS615_NODEJS" containing files: "test-node", "1_basic.js", "2_functions.js", and "app.js".
- CODE EDITOR**: The active tab is "2_functions.js" which contains the following code:

```
1 function sayHi(){
2     console.log('Hi');
3 }
4 sayHi();
```
- TERMINAL**: The terminal tab is active and shows the output of running the script:

```
● jojojustine@MacBookAir CS615_nodejs % node 1_basic.js
2 seconds have passed!
3 seconds have passed
4 seconds have passed!
5 seconds have passed!
6 seconds have passed!
● jojojustine@MacBookAir CS615_nodejs % node 1_basic.js
/Users/jojojustine/Downloads/Study/Semester2/Internetsolutions/git/CS615C/Assignment5/CS615_nodejs
/Users/jojojustine/Downloads/Study/Semester2/Internetsolutions/git/CS615C/Assignment5/CS615_nodejs/1_basic.js
2 seconds have passed!
3 seconds have passed
4 seconds have passed!
5 seconds have passed!
6 seconds have passed!
● jojojustine@MacBookAir CS615_nodejs % node 2_functions.js
Hi
● jojojustine@MacBookAir CS615_nodejs %
```

The screenshot shows the Visual Studio Code interface with the following details:

- EXPLORER** sidebar: Shows a folder named "CS615_NODEJS" containing files: "test-node", "1_basic.js", "2_functions.js" (selected), and "app.js".
- CODE EDITOR**: The active tab is "2_functions.js". The code is as follows:

```
//function sayHi(){
//    console.log('Hi');
//}
// sayHi();
var sayBye = function(){
    console.log('Bye');
};
sayBye();
```

- TERMINAL**: Displays the output of running Node.js scripts.

 - Output of "1_basic.js":
3 seconds have passed!
4 seconds have passed!
6 seconds have passed!
 - Output of "2_functions.js":
● jojojustine@MacBookAir CS615_nodejs % node 1_basic.js
/Users/jojojustine/Downloads/Study/Semester2/Internetsolutions/git/CS615C/Assignment5/CS615_nodejs
/Users/jojojustine/Downloads/Study/Semester2/Internetsolutions/git/CS615C/Assignment5/CS615_nodejs/1_basic.js
2 seconds have passed!
3 seconds have passed
4 seconds have passed!
6 seconds have passed!
 - Output of "2_functions.js":
● jojojustine@MacBookAir CS615_nodejs % node 2_functions.js
Hi
● jojojustine@MacBookAir CS615_nodejs % Bye
○ jojojustine@MacBookAir CS615_nodejs %

- STATUS BAR**: Shows "Ln 4 Col 3" and "Spaces: 4" along with a "UIT" indicator.

The screenshot shows a dark-themed instance of Visual Studio Code (VS Code) with the following interface elements:

- Explorer View:** On the left, it shows a tree structure for the folder "CS615_NODEJS". The files listed are "test-node", "1_basic.js", "2_functions.js", and "app.js".
- Editor View:** The main area displays the content of the file "2_functions.js". The code is as follows:

```
//function sayHi(){
//    console.log('Hi');
//}
//sayHi();

var sayBye = function(){
    console.log('Bye');
};
sayBye();

function callFunction(fun){
    fun();
}
callFunction(sayBye);
```

- Terminal View:** At the bottom, the terminal window shows the output of running Node.js scripts:
- Execution of "1_basic.js":
jоjojustine@MacBookAir CS615_nodejs % node 1_basic.js
/Users/jоjojustine/Downloads/Study/Semester2/Internetsolutions/git/CS615C/Assignment5/CS615_nodejs
/Users/jоjojustine/Downloads/Study/Semester2/Internetsolutions/git/CS615C/Assignment5/CS615_nodejs/1_basic.js
2 seconds have passed!
3 seconds have passed
4 seconds have passed!
6 seconds have passed!
- Execution of "2_functions.js":
jоjojustine@MacBookAir CS615_nodejs % node 2_functions.js
Hi
jоjojustine@MacBookAir CS615_nodejs % node 2_functions.js
Bye
jоjojustine@MacBookAir CS615_nodejs % node 2_functions.js
Bye
Bye

The screenshot shows the VS Code interface with the title bar "CS615_nodejs". The Explorer sidebar on the left lists files under "CS615_NODEJS" including "test-node", "1_basic.js", "2_functions.js", "3_math.js", and "app.js". The terminal tab at the top has tabs for "Welcome", "app.js", "1_basic.js", "2_functions.js", "3_math.js" (which is currently active), and "hello.js". The main terminal window displays the following command-line session:

```
6 seconds have passed!
● jojojustine@MacBookAir CS615_nodejs % node 2_functions.js
Hi
● jojojustine@MacBookAir CS615_nodejs % node 2_functions.js
Bye
● jojojustine@MacBookAir CS615_nodejs % node 2_functions.js
Bye
Bye
● jojojustine@MacBookAir CS615_nodejs % 3_math.js
zsh: command not found: 3_math.js
● jojojustine@MacBookAir CS615_nodejs % 3_math.js
zsh: command not found: 3_math.js
● jojojustine@MacBookAir CS615_nodejs % node 3_math.js
There are 3 elements in this array.
○ jojojustine@MacBookAir CS615_nodejs %
```

The screenshot shows the VS Code interface with the title bar "CS615_nodejs". The Explorer sidebar on the left lists files under "CS615_NODEJS" including "test-node", "1_basic.js", "2_functions.js", "3_math.js", and "app.js". The terminal tab at the top has tabs for "Welcome", "app.js" (which is currently active), "1_basic.js", "2_functions.js", "3_math.js", and "hello.js". The main terminal window displays the following command-line session, showing a "MODULE_NOT_FOUND" error:

```
at require (node:internal/modules/helpers:175:16)
at Object. (<file>:1:1)
at Module._compile (node:internal/modules/cjs/loader:1734:14) {
  code: 'MODULE_NOT_FOUND',
  requireStack: [
    <file>
  ]
}
Node.js v23.9.0
● jojojustine@MacBookAir CS615_nodejs % node app.js
Hello World!
There are 3 elements in this array.
There are 3 elements in this array.
○ jojojustine@MacBookAir CS615_nodejs %
```

The screenshot shows the Visual Studio Code (VS Code) interface with a dark theme. The left sidebar has icons for Explorer, Search, Open, and Terminal. The Explorer view shows a folder named 'CS615_NODEJS' containing files: 'test-node', '1_basic.js', '2_functions.js', '3_math.js', and 'app.js'. The 'app.js' file is selected and open in the main editor tab. The code in 'app.js' is:

```
1 console.log("Hello World!");
2 var counter = require('./3_math.js');
3 console.log(counter(['apple', 'pear', 'cherry']));
4 var math = require('./3_math');
5 console.log(math.counter(['Shaun','Crystal', 'Ryu']));
6 console.log(math.adder(math.pi, 10));
7
```

The terminal tab at the bottom shows the output of running the script:

```
There are 3 elements in this array.
There are 3 elements in this array.
● jojojustine@MacBookAir CS615_nodejs % node 3_math.js
There are 3 elements in this array.
● jojojustine@MacBookAir CS615_nodejs % node app.js
Hello World!
There are 3 elements in this array.
There are 3 elements in this array.
● jojojustine@MacBookAir CS615_nodejs % node app.js
Hello World!
There are 3 elements in this array.
There are 3 elements in this array.
There are 3 elements in this array.
The sum of the 2 numbers is 13.1415
○ jojojustine@MacBookAir CS615_nodejs %
```

The screenshot shows the VS Code interface with the title bar "CS615_nodejs". The Explorer sidebar on the left shows a folder "CS615_NODEJS" containing files: test-node, JS 1_basic.js, JS 2_functions.js, JS 3_math.js, and JS app.js. The active tab in the center is "app.js" with the following code:

```
1 console.log("Hello World!");
2 var counter = require('./3_math.js');
3 console.log(counter('apple', 'pear', 'cherry'));
4 var math = require('./3_math');
5 console.log(math.counter('Shaun', 'Crystal', 'Ryu'));
6 console.log(math.adder(math.pi, 10));
```

The terminal at the bottom shows the output of running "node app.js":

```
Node.js v23.9.0
● jojojustine@MacBookAir CS615_nodejs % node app.js
Hello World!
```

The Problems, Output, Debug Console, and Ports tabs are visible at the bottom.

The screenshot shows the VS Code interface with the title bar "CS615_nodejs". The Explorer sidebar on the left shows a folder "CS615_NODEJS" containing files: test-node, JS 1_basic.js, JS 2_functions.js, JS 3_math.js, JS 4_fs.js, JS app.js, and a file icon labeled "readMe.txt". The active tab in the center is "4_fs.js" with the following code:

```
1 var fs = require('fs');
2 //var readMe = fs.readFileSync(...);
3 var readMe = fs.readFileSync('readMe.txt', 'utf8');
4 console.log(readMe);
```

The terminal at the bottom shows the output of running "node 4_fs.js":

```
Node.js v23.9.0
● jojojustine@MacBookAir CS615_nodejs % node 4_fs.js
In the distant land of Blorpagonia, Tuesdays were reserved for synchronized llama dancing. The llamas, dressed in neon bow ties and glittery socks, gathered around the Great Jellybean Fountain, where Elder Zog read poetry from a banana peel.
```

The Problems, Output, Debug Console, and PORTS tabs are visible at the bottom.

The screenshot shows the VS Code interface with the title bar "CS615_nodejs". The Explorer sidebar on the left lists files in the "CS615_NODEJS" folder, including "app.js", "readMe.txt", "4_fs.js", "1_basic.js", "2_functions.js", "3_math.js", and "hello.js". The "writeMe.txt" file is selected and its content is displayed in the main editor area:

```
1 In the distant land of Blorpagonia, Tuesdays were reserved for synchronized llama dancing. The llamas, dressed in neon bow ties and g
```

The Terminal tab at the bottom shows the command "node 4_fs.js" being run, and the output is:

```
at node:internal/main/run_main_module:33:47 {
  errno: -2,
  code: 'ENOENT',
  syscall: 'open',
  path: 'readMe.txt'
}

Node.js v23.9.0
● jojojustine@MacBookAir CS615_nodejs % node 4_fs.js
In the distant land of Blorpagonia, Tuesdays were reserved for synchronized llama dancing. The llamas, dressed in neon bow ties and glittery socks, gathered arou
nd the Great Jellybean Fountain, where Elder Zog read poetry from a banana peel.
```

The screenshot shows the VS Code interface with the title bar "CS615_nodejs". The Explorer sidebar on the left lists files in the "CS615_NODEJS" folder, including "app.js", "readMe.txt", "4_fs.js", "1_basic.js", "2_functions.js", "3_math.js", and "hello.js". The "4_fs.js" file is selected and its content is displayed in the main editor area:

```
1 var fs = require('fs');
2 //var readMe = fs.readFileSync(...);
3 var readMe = fs.readFileSync('readMe.txt', 'utf8');
4 //console.log(readMe);
5 //fs.writeFileSync('writeMe.txt', readMe);
6 fs.readFile('readMe.txt', 'utf8', function(err, data){
7   console.log('File read!');
8 });
9 console.log('After Reading.');

Node.js v23.9.0
● jojojustine@MacBookAir CS615_nodejs % node 4_fs.js
In the distant land of Blorpagonia, Tuesdays were reserved for synchronized llama dancing. The llamas, dressed in neon bow ties and glittery socks, gathered arou
nd the Great Jellybean Fountain, where Elder Zog read poetry from a banana peel.
```

The Terminal tab at the bottom shows the command "node 4_fs.js" being run again, and the output is identical to the previous screenshot.

The screenshot shows the VS Code interface with the title bar "CS615_nodejs". The Explorer sidebar on the left lists files: CS615_NODEJS, test-node, 1_basic.js, 2_functions.js, 3_math.js, 4_fs.js (which is selected), app.js, and writeMe.txt. The main editor area displays the contents of 4_fs.js:

```
1 var fs = require('fs');
2 //var readMe = fs.readFileSync(...);
3 var readMe = fs.readFileSync('readMe.txt', 'utf8');
4 //console.log(readMe);
5 //fs.writeFileSync('writeMe.txt', readMe);
6 //fs.readFile('readMe.txt', 'utf8', function(err, data){
7 //    console.log('File read!');
8 //});
9 //    console.log('After Reading.');
10
11 fs.readFile('readMe.txt', 'utf8', function(err, data){
12     console.log('File read!');
13     fs.writeFile('writeMe.txt', data, function(err, data){
14         console.log('File written!');
15     });
16 });
17
18 console.log('This is some code after the read-write code');
```

The terminal tab at the bottom shows the output of running node 4_fs.js:

```
Node.js v23.9.0
jojojustine@MacBookAir CS615_nodejs % node 4_fs.js
In the distant land of Blorpagonia, Tuesdays were reserved for synchronized llama dancing. The llamas, dressed in neon bow ties and glittery socks, gathered around the Great Jellybean Fountain, where Elder Zog read poetry from a banana peel.
jojojustine@MacBookAir CS615_nodejs % node 4_fs.js
In the distant land of Blorpagonia, Tuesdays were reserved for synchronized llama dancing. The llamas, dressed in neon bow ties and glittery socks, gathered around the Great Jellybean Fountain, where Elder Zog read poetry from a banana peel.
jojojustine@MacBookAir CS615_nodejs % node 4_fs.js
After reading.
File read!
jojojustine@MacBookAir CS615_nodejs % node 4_fs.js
This is some code after the read-write code
File read!
File written!
jojojustine@MacBookAir CS615_nodejs %
```

Part 3

The screenshot shows the VS Code interface with the title bar "CS615_nodejs". The Explorer sidebar on the left lists files: CS615_NODEJS, test-node, 1_basic.js, 2_functions.js, 3_math.js, 4_fs.js, 5_httpserver.js (which is selected), app.js, and writeMe.txt. The main editor area displays the contents of 5_httpserver.js:

```
1 var http = require('http');
2 var server = http.createServer();
3 server.listen(3000, '127.0.0.1');
4 console.log('Listening to port 3000 ...');
```

The terminal tab at the bottom shows the output of running node 5_httpserver.js:

```
In the distant land of Blorpagonia, Tuesdays were reserved for synchronized llama dancing. The llamas, dressed in neon bow ties and glittery socks, gathered around the Great Jellybean Fountain, where Elder Zog read poetry from a banana peel.
jojojustine@MacBookAir CS615_nodejs % node 4_fs.js
In the distant land of Blorpagonia, Tuesdays were reserved for synchronized llama dancing. The llamas, dressed in neon bow ties and glittery socks, gathered around the Great Jellybean Fountain, where Elder Zog read poetry from a banana peel.
jojojustine@MacBookAir CS615_nodejs % node 4_fs.js
After reading.
File read!
jojojustine@MacBookAir CS615_nodejs % node 4_fs.js
This is some code after the read-write code
File read!
File written!
jojojustine@MacBookAir CS615_nodejs % node 5_httpserver.js
Listening to port 3000 ...
```

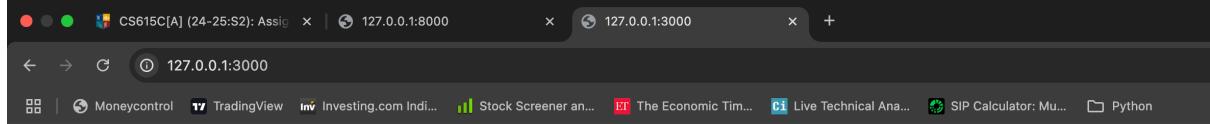
The screenshot shows the Visual Studio Code interface. The Explorer sidebar on the left lists files in a folder named 'CS615_NODEJS'. The code editor tab bar includes 'app.js', 'readMe.txt', '4_fs.js', '5_httpserver.js' (which is currently active), 'writeMe.txt', '1_basic.js', '2_functions.js', '3_math.js', and 'hello.j'. The main code editor window displays the content of '5_httpserver.js':

```
1 var http = require('http');
2 //var server = http.createServer();
3 var server = http.createServer(function(req, res){
4   console.log('Request was made: ' + req.url);
5   res.writeHead(200, {'Content-Type' : 'text/html'});
6   res.end('Hello students\n');
7 });
8 server.listen(3000, '127.0.0.1');
9 console.log('Listening to port 3000 ...');
```

The Terminal tab bar at the bottom shows command-line history:

- After Reading.
- File read!
- This is some code after the read-write code
- File written!
- Listening to port 3000 ...
- Request was made: /
- Request was made: /favicon.ico

The status bar at the bottom right indicates: Ln 7, Col 8 | Spaces: 4 | UTF-8 | LF | JavaScript | Go Live.



Hello students!

The screenshot shows a VS Code interface with the following details:

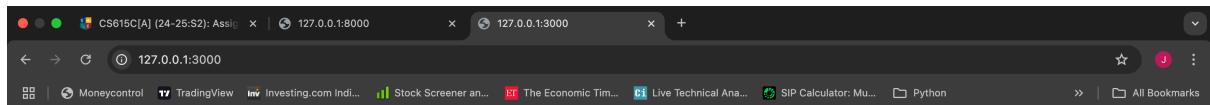
- Explorer View:** Shows files in the CS615_NODEJS folder, including app.js, readMe.txt, 4_fs.js, 5_httpserver.js, 6_streams.js, writeMe.txt, 1_basic.js, 2_functions.js, 3_math.js, 4_fs.js, 5_httpserver.js, and 6_streams.js.
- Editor View:** The 6_streams.js file is open, displaying code for reading and writing streams. The code uses `http` and `fs` modules to handle a file named `readMe.txt`.
- Terminal View:** The terminal window shows the execution of three scripts:
 - `node 5_httpserver.js`: Listening on port 3000.
 - `node 6_streams.js`: Listening on port 3000, handling a request for `/favicon.ico`.
 - `node 6_streams.js`: A second instance, showing a message about synchronized llama dancing.
- Status Bar:** Shows the current file as `6_streams.js`, the line number as 11, and the column number as 36. It also displays tabs for Spaces: 4, UTF-8, LF, JavaScript, and Go Live.

The screenshot shows the VS Code interface with the following details:

- EXPLORER** sidebar: Shows files in the CS615_NODEJS folder: test-node, 1_basic.js, 2_functions.js, 3_math.js, 4_fs.js, 5_httpserver.js, 6_streams.js, app.js, readMe.txt, and writeMe.txt.
- CODE EDITOR**: The active file is 6_pipeServer.js. The code defines a server that reads from a file ('readMe.txt') and pipes it to the response object ('res').

```
var http = require('http');
var fs = require('fs');
var server = http.createServer(function(req, res){
    console.log('Request was made: ' + req.url);
    var myReadStream = fs.createReadStream(__dirname + '/readMe.txt', 'utf8');
    myReadStream.pipe(res);
});
server.listen(3000, '127.0.0.1');
console.log('Listening to port 3000...');
```
- TERMINAL**: Shows the command-line output of running the script.

```
Request was made: /favicon.ico
Request was made: /
Request was made: /favicon.ico
^C
obj@obj-OptiPlex-5070:~/Documents/cs615$ node 6_pipeServer.js
Listening to port 3000...
Request was made: /
Request was made: /favicon.ico
```



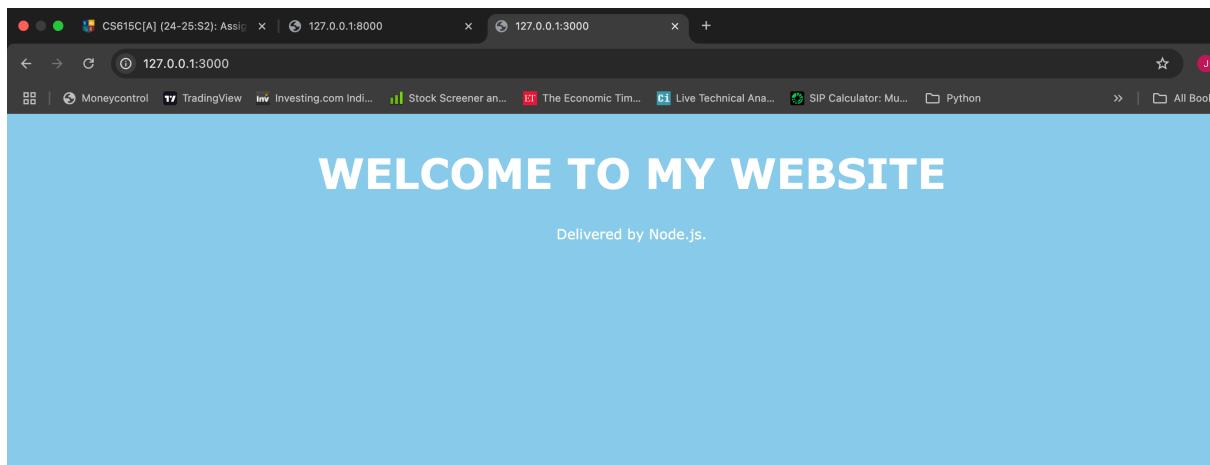
In the distant land of Blorpagonia, Tuesdays were reserved for synchronized llama dancing. The llamas, dressed in neon bow ties and glittery socks, gathered around the Great Jellybean Fountain, where Elder Zog read poetry from a banana peel.

```
<html>
  <head>
    <style>
      body {background: #skyblue; font-family: verdana; color: #fffff; padding: 30px;}
      h1 {font-size: 48px; text-transform: uppercase; text-align: center;}
      p {font-size: 16px; text-align: center;}
    </style>
  </head>
  <body>
    <h1>Welcome to my Website</h1>
    <p>Delivered by Node.js.</p>
  </body>
</html>
```

```
var http = require('http');
var fs = require('fs');
var server = http.createServer(function(req, res){
  console.log('request was made: ' + req.url);
  res.writeHead(200, {'Content-Type': 'text/html'});
  var myReadStream = fs.createReadStream(__dirname + '/index.html', 'utf8');
  myReadStream.pipe(res);
});
server.listen(3000, '127.0.0.1');
console.log('Listening to port 3000...');
```

The terminal below shows the command being run and the resulting output:

```
listing@MacBookAir CS615_nodejs % node 7_servingHTML.js
Listening to port 3000...
request was made: /favicon.ico
^C
listing@MacBookAir CS615_nodejs % node 7_servingHTML.js
Listening to port 3000...
request was made: /favicon.ico
request was made: /
request was made: /favicon.ico
^C
listing@MacBookAir CS615_nodejs % node 7_servingHTML.js
Listening to port 3000...
request was made: /favicon.ico
```



Part 4

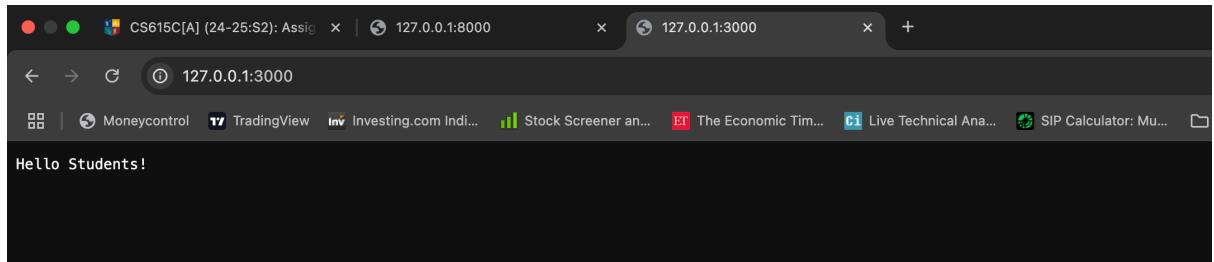
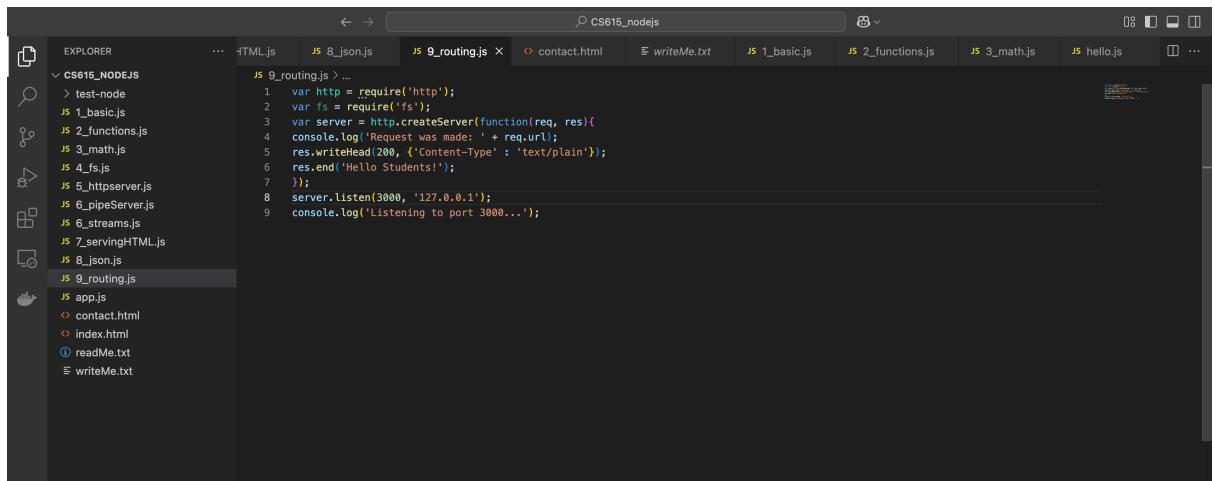
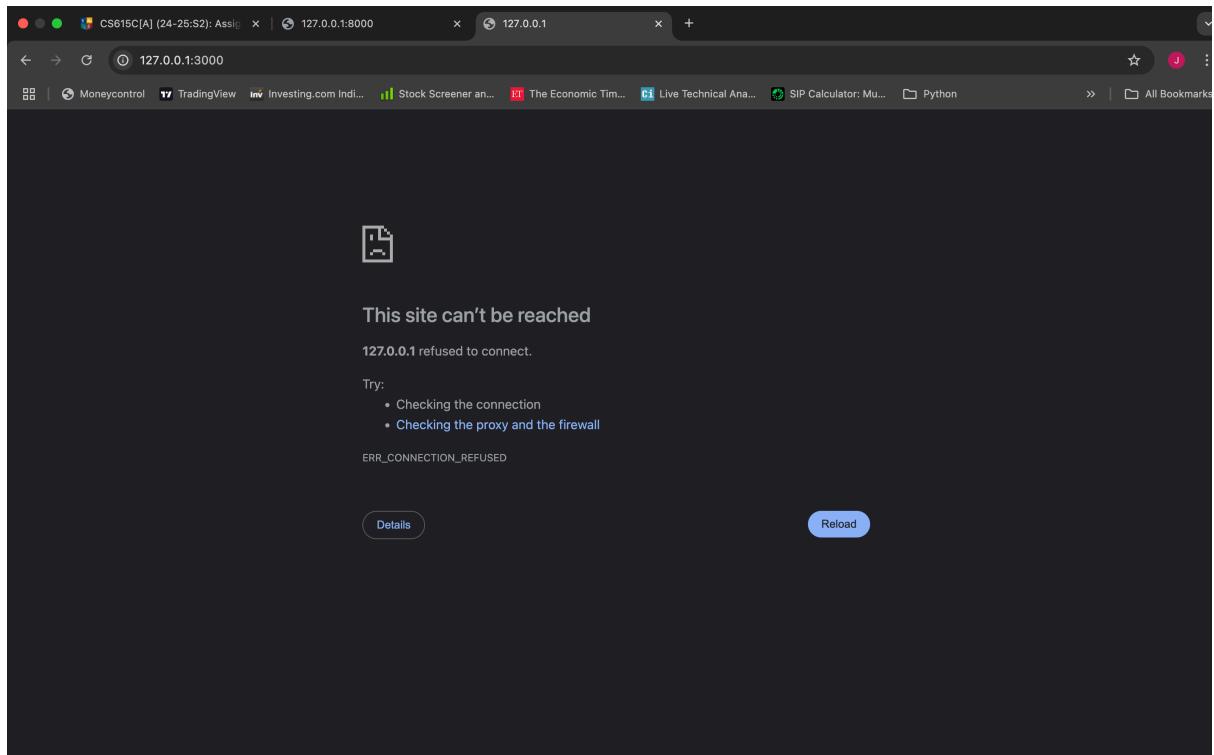
A screenshot of the Visual Studio Code (VS Code) interface. The left sidebar shows a file tree with files like "readMe.txt", "4_fs.js", "5_httpserver.js", "6_streams.js", "6_pipeServer.js", "7_servingHTML.js", "8_json.js", "app.js", "index.html", "readMe.txt", and "writeMe.txt". The main editor area displays the "8_json.js" file with the following code:

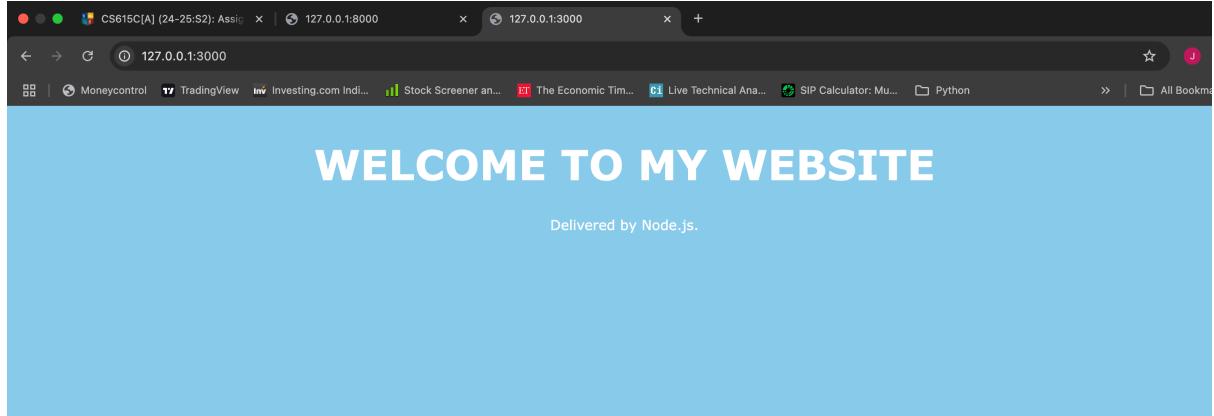
```
var http = require('http');
var fs = require('fs');
var server = http.createServer(function(req, res){
  console.log('request was made: ' + req.url);
  res.writeHead(200, {'Content-Type': 'application/json'});
  myObject = {
    name: 'Paul',
    job: 'student',
    age: 25
  };
  // res.end(myObj); // does not work
  res.end(JSON.stringify(myObj));
});
server.listen(3000, '127.0.0.1');
console.log('Listening to port 3000...');
```

The bottom right terminal tab shows the output of running the script: "Listening to port 3000..." followed by a stack trace for a ReferenceError.

```
Listening to port 3000...
request was made: /
/Users/jojojustine/Downloads/Study/Semester2/Internetsolutions/git/CS615C/Assignment5/CS615_nodejs/8_json.js:12
  res.end(JSON.stringify(myObj));
^

ReferenceError: myObj is not defined
  at Server.<anonymous> (/Users/jojojustine/Downloads/Study/Semester2/Internetsolutions/git/CS615C/Assignment5/CS615_nodejs/8_json.js:12:24)
  at Server.emit (node:events:507:28)
  at parserOnIncoming (node:HTTPParser:1153:12)
  at HTTPParser.parserOnHeadersComplete (node:HTTPParser:117:17)
```





The screenshot shows a Node.js application running in a browser. The browser window title is "CS615_nodejs". The address bar shows the URL "127.0.0.1:3000". The page content is a simple "WELCOME TO MY WEBSITE" in large white text on a light blue background, with the subtitle "Delivered by Node.js." below it.

At the top of the browser window, there are several tabs: "HTML.js", "JS 8_json.js", "JS 9_routing.js" (which is the active tab), "contact.html", "writeMe.txt", "JS 1_basic.js", "JS 2_functions.js", "JS 3_math.js", and "JS hello.js".

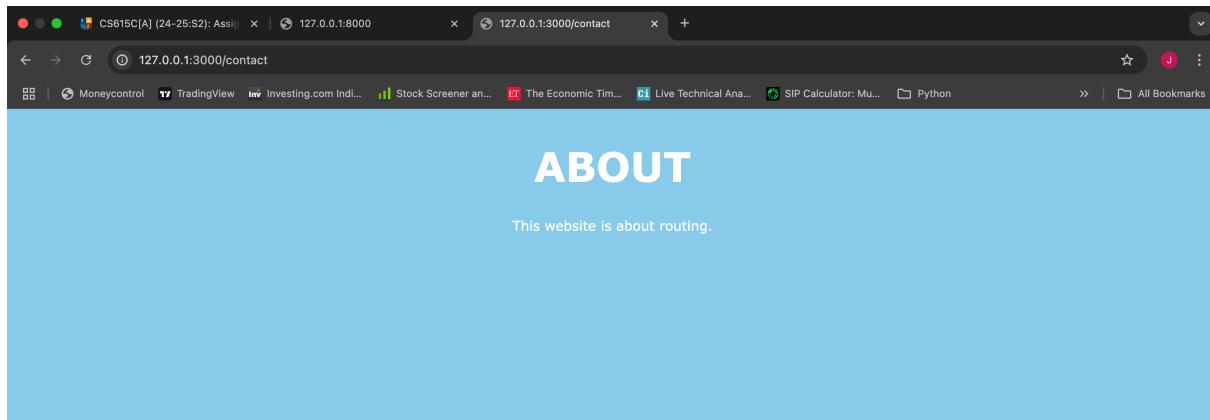
The browser's status bar at the bottom shows "Ln 10, Col 3 Spaces: 4 UTF-8 LF ⚡ JavaScript Go".

The main content area of the browser displays the website's content.

```
var http = require('http');
var fs = require('fs');

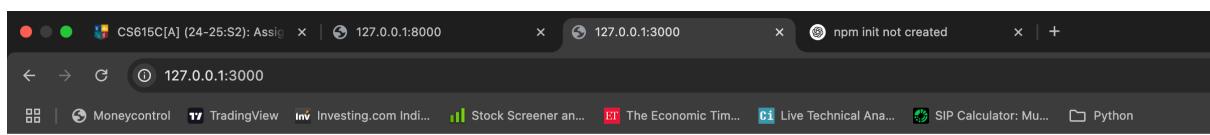
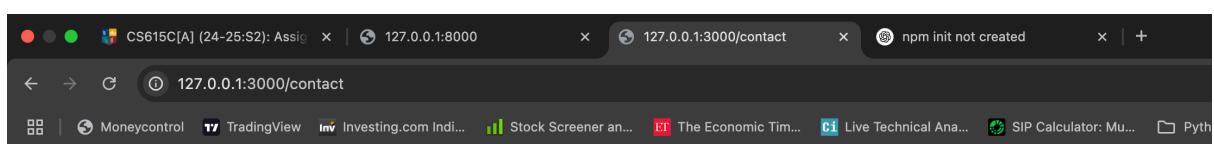
var server = http.createServer(function(req, res){
    if (req.url === '/home' || req.url === '/') {
        res.writeHead(200, {'Content-Type': 'text/html'});
        fs.createReadStream(__dirname + '/index.html').pipe(res);
    } else if (req.url === '/contact') {
        res.writeHead(200, {'Content-Type': 'text/html'});
        fs.createReadStream(__dirname + '/contact.html').pipe(res);
    }
});

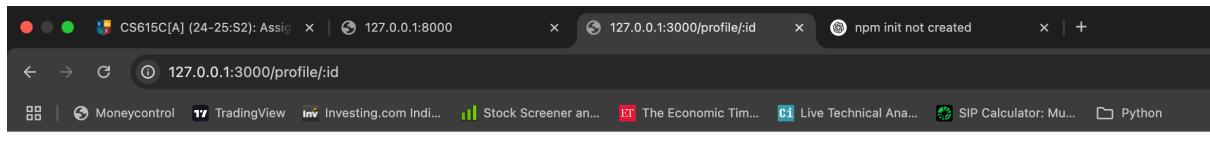
server.listen(3000, '127.0.0.1');
console.log('Listening to port 3000...');
```



```
expressApp > JS app.js > ...
1 var express = require('express');
2 var app = express();
3 //
4 //
5 // We will handle requests here
6 app.listen(3000);
7 app.get('/', function(req, res){
8   res.send('This is the homepage.');
9 });
10 app.get('/contact', function(req, res){
11   res.send('This is the contact page.');
12 });
13 app.get('/profile/:id', function(req, res){
14   res.send('You requested to see profile with the id of ' +
15   req.params.id);
16 });

Is this OK? (yes) yes
● jojojustine@MacBookAir expressApp % npm install express -save
added 69 packages, and audited 70 packages in 2s
14 packages are looking for funding
  run `npm fund` for details
found 0 vulnerabilities
○ jojojustine@MacBookAir expressApp % node app.js
```





You requested to see profile with the id of :id