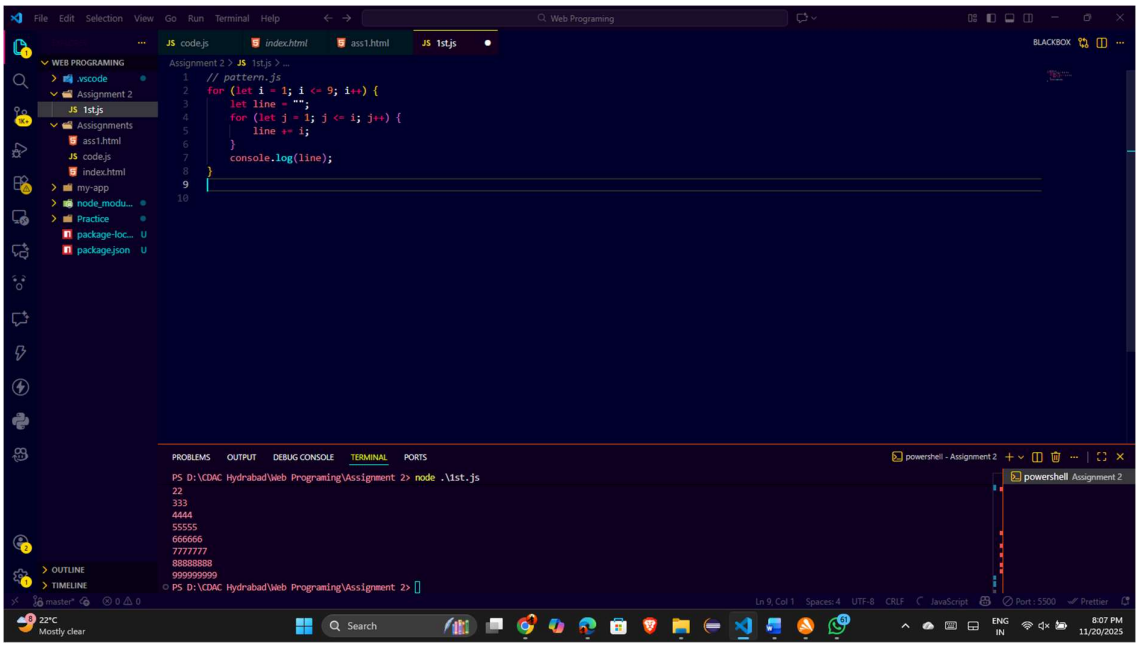
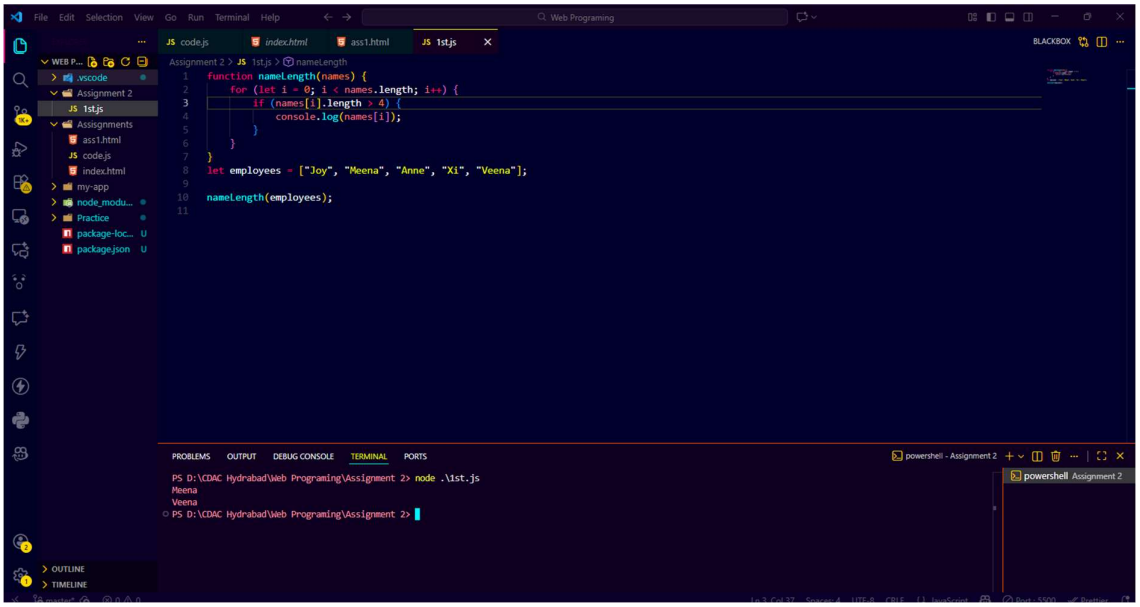
Assignment 02

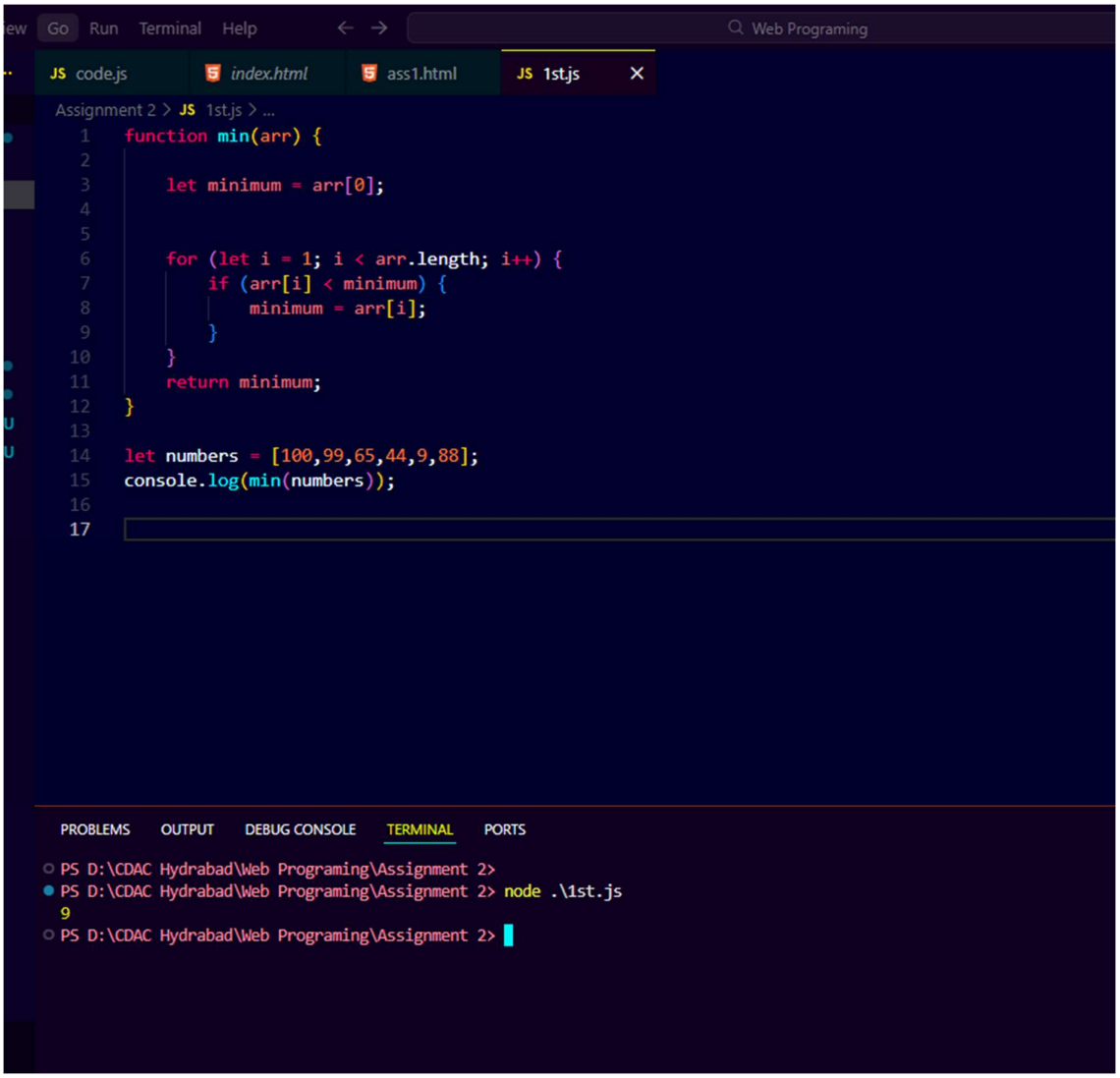
Q1)



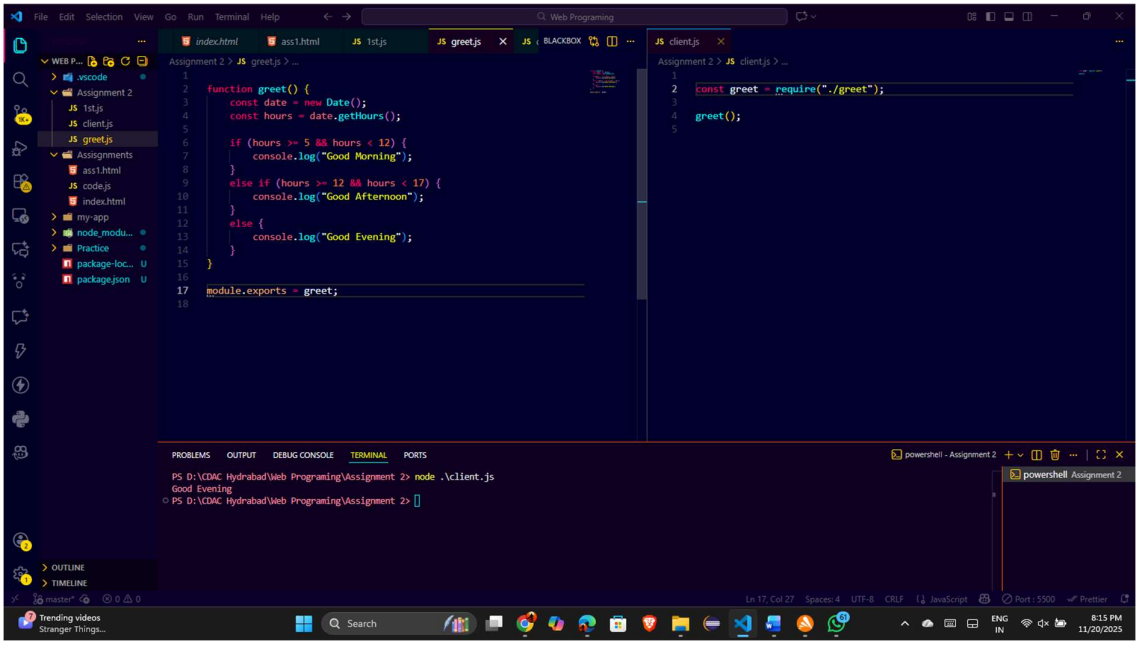
Q2)



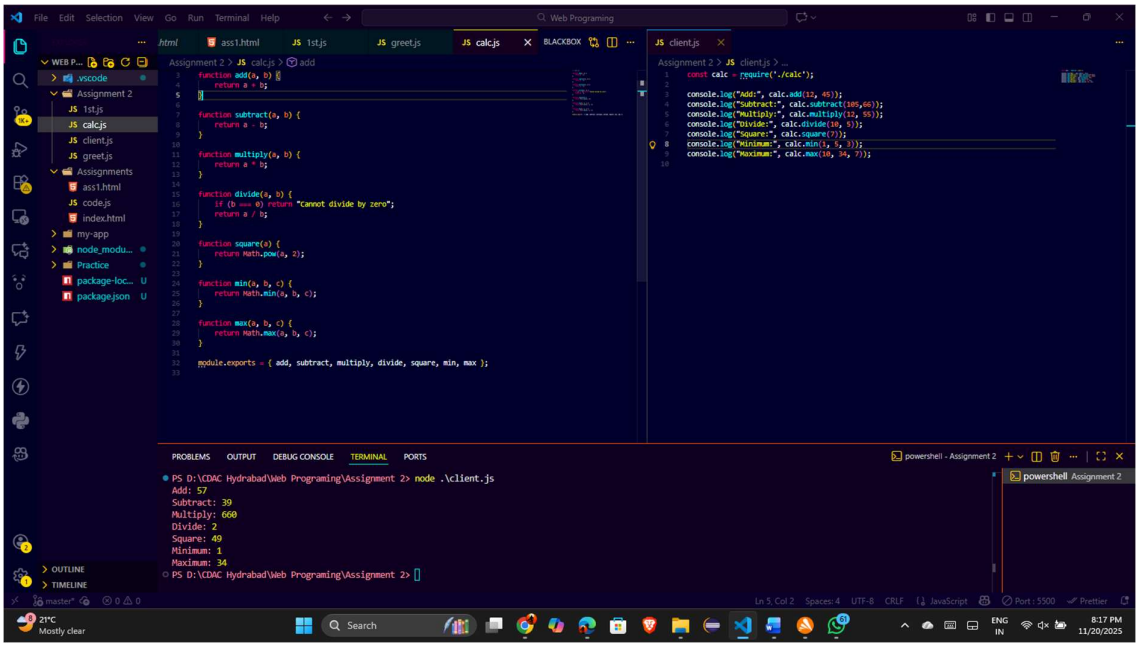
Q3)


Go Run Terminal Help
Web Programing
code.js
index.html
ass1.html
1st.js
Assignment 2 > 1st.js > ...
function min(arr) {
let minimum = arr[0];
for (let i = 1; i < arr.length; i++) {
if (arr[i] < minimum) {
minimum = arr[i];
}
}
return minimum;
}
let numbers = [100,99,65,44,9,88];
console.log(min(numbers));
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
PS D:\CDAC Hydrabad\Web Programing\Assignment 2>
PS D:\CDAC Hydrabad\Web Programing\Assignment 2> node .\1st.js
9
PS D:\CDAC Hydrabad\Web Programing\Assignment 2>

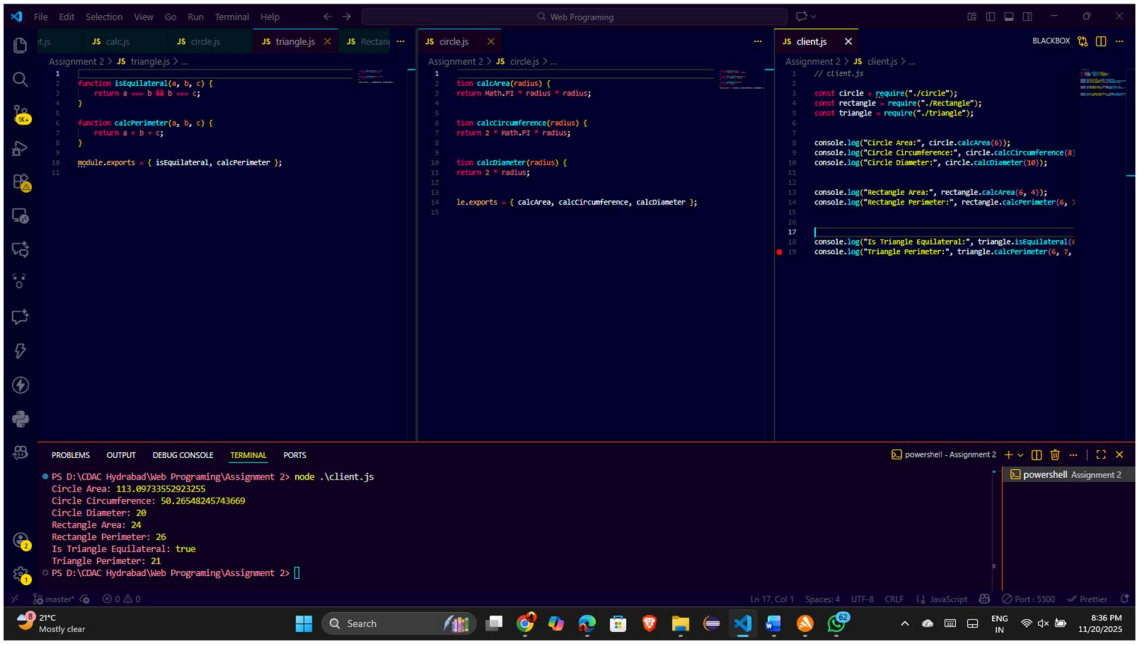
Q4)


function greet() {
const date = new Date();
const hours = date.getHours();
if (hours >= 5 && hours < 12) {
console.log("Good Morning");
}
else if (hours >= 12 && hours < 17) {
console.log("Good Afternoon");
}
else {
console.log("Good Evening");
}
}
module.exports = greet;
const greet = require("./greet");
greet();
PS D:\CDAC Hydrabad\Web Programing\Assignment 2> node .\client.js
Good Evening
PS D:\CDAC Hydrabad\Web Programing\Assignment 2>

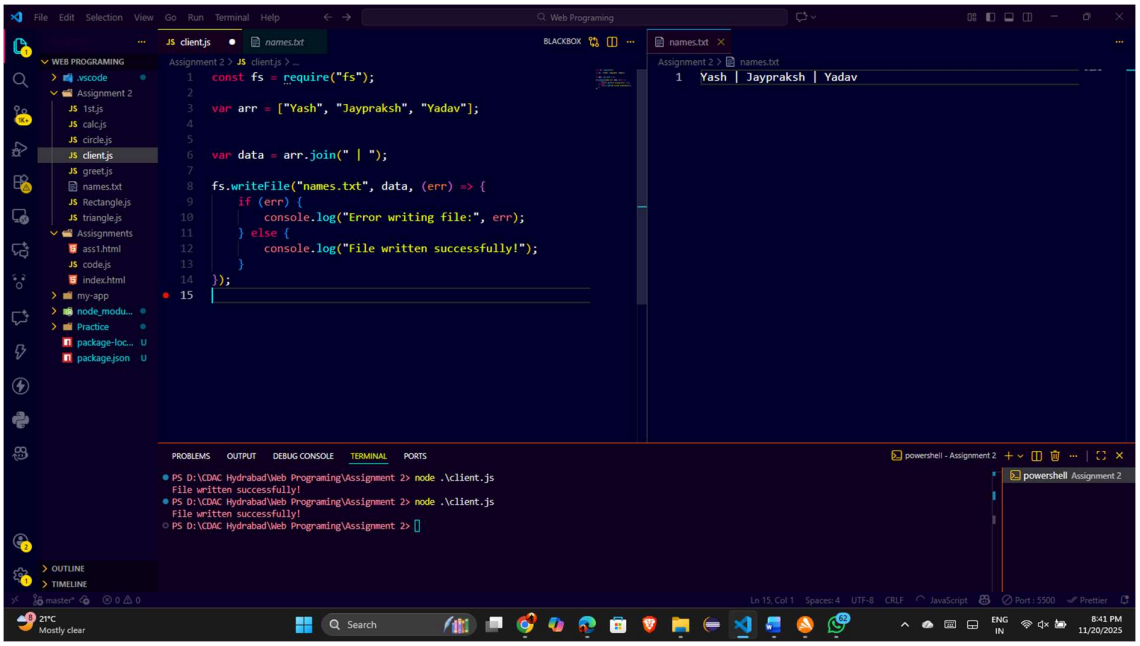
Q5)


function add(a, b) {
return a + b;
}
function subtract(a, b) {
return a - b;
}
function multiply(a, b) {
return a * b;
}
function divide(a, b) {
if (b === 0) return "Cannot divide by zero";
return a / b;
}
function square(a) {
return Math.pow(a, 2);
}
function min(a, b, c) {
return Math.min(a, b, c);
}
function max(a, b, c) {
return Math.max(a, b, c);
}
module.exports = { add, subtract, multiply, divide, square, min, max };
const calc = require('./calc');
console.log("Add:", calc.add(12, 45));
console.log("Subtract:", calc.subtract(105,66));
console.log("Multiply:", calc.multiply(11, 55));
console.log("Divide:", calc.divide(10, 5));
console.log("Square:", calc.square(7));
console.log("Minimum:", calc.min(1, 5, 3));
console.log("Maximum:", calc.max(10, 34, 7));
PS D:\CDAC Hydrabad\Web Programing\Assignment 2> node .\client.js
Add: 57
Subtract: 39
Multiply: 660
Divide: 2
Square: 49
Minimum: 1
Maximum: 34
PS D:\CDAC Hydrabad\Web Programing\Assignment 2>

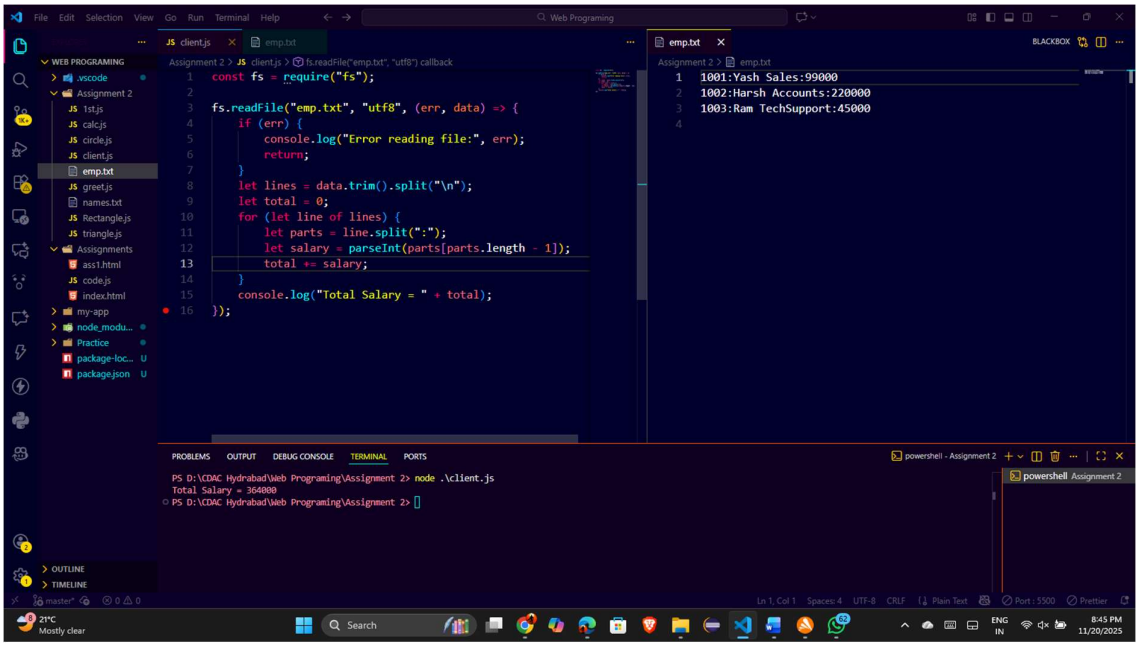
Q6)


PS D:\CDAC Hydrabad\Web Programing\Assignment 2> node .\client.js
Circle Area: 113.09733552923255
Circle Circumference: 50.26548245743669
Circle Diameter: 20
Rectangle Area: 24
Rectangle Perimeter: 26
Is Triangle Equilateral: true
Triangle Perimeter: 21
PS D:\CDAC Hydrabad\Web Programing\Assignment 2>

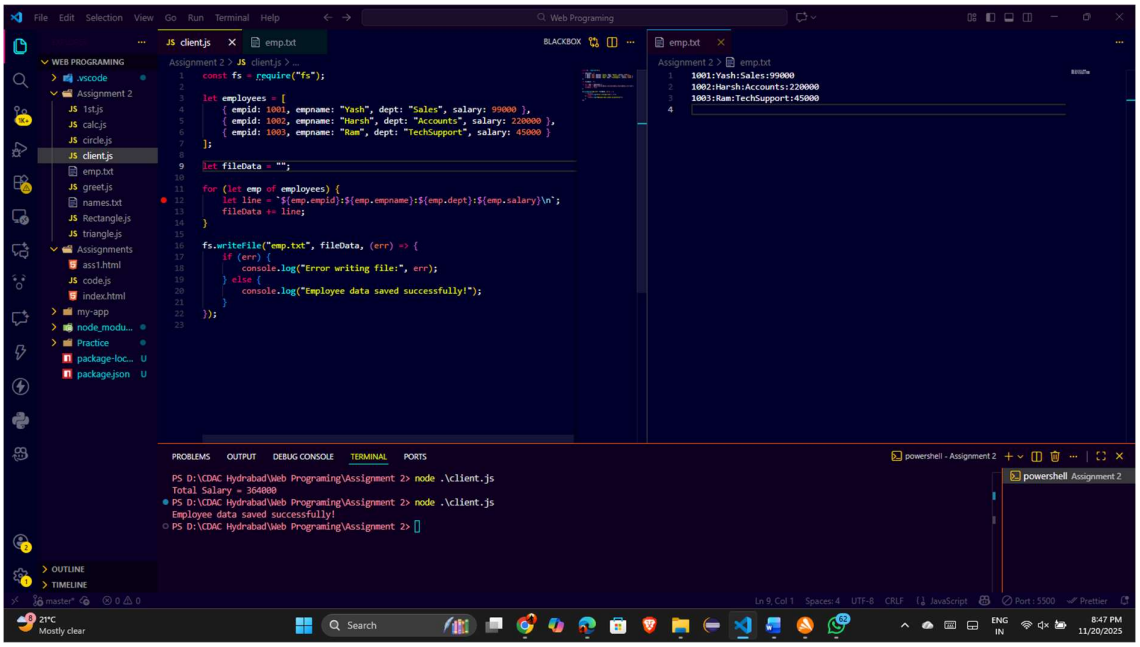
Q7)


const fs = require("fs");
var arr = ["Yash", "Jaypraksh", "Yadav"];
var data = arr.join(" | ");
fs.writeFile("names.txt", data, (err) => {
if (err) {
console.log("Error writing file:", err);
} else {
console.log("File written successfully!");
}
});
Yash | Jaypraksh | Yadav
PS D:\CDAC Hydrabad\Web Programing\Assignment 2> node .\client.js
File written successfully!
PS D:\CDAC Hydrabad\Web Programing\Assignment 2> node .\client.js
File written successfully!
PS D:\CDAC Hydrabad\Web Programing\Assignment 2>

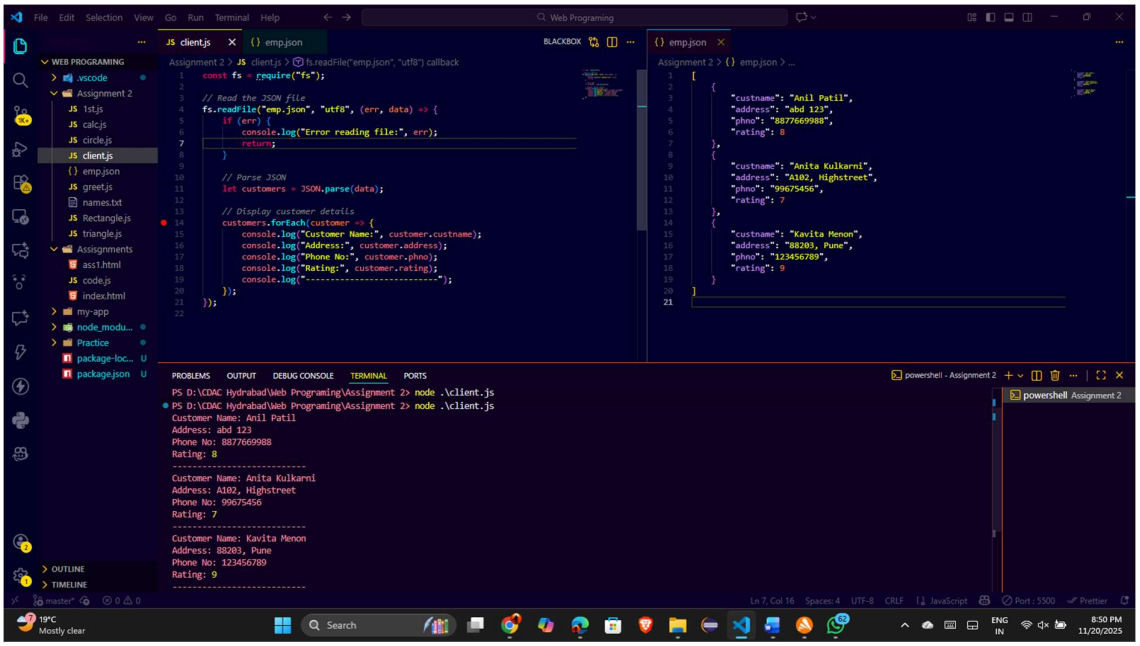
Q8)


const fs = require("fs");
fs.readFile("emp.txt", "utf8", (err, data) => {
if (err) {
console.log("Error reading file:", err);
return;
}
let lines = data.trim().split("\n");
let total = 0;
for (let line of lines) {
let parts = line.split(":");
let salary = parseInt(parts[parts.length - 1]);
total += salary;
}
console.log("Total Salary = " + total);
});
1001:Yash Sales:99000
1002:Harsh Accounts:220000
1003:Ram TechSupport:45000
PS D:\CDAC Hydrabad\Web Programing\Assignment 2> node .\client.js
Total Salary = 364000
PS D:\CDAC Hydrabad\Web Programing\Assignment 2>

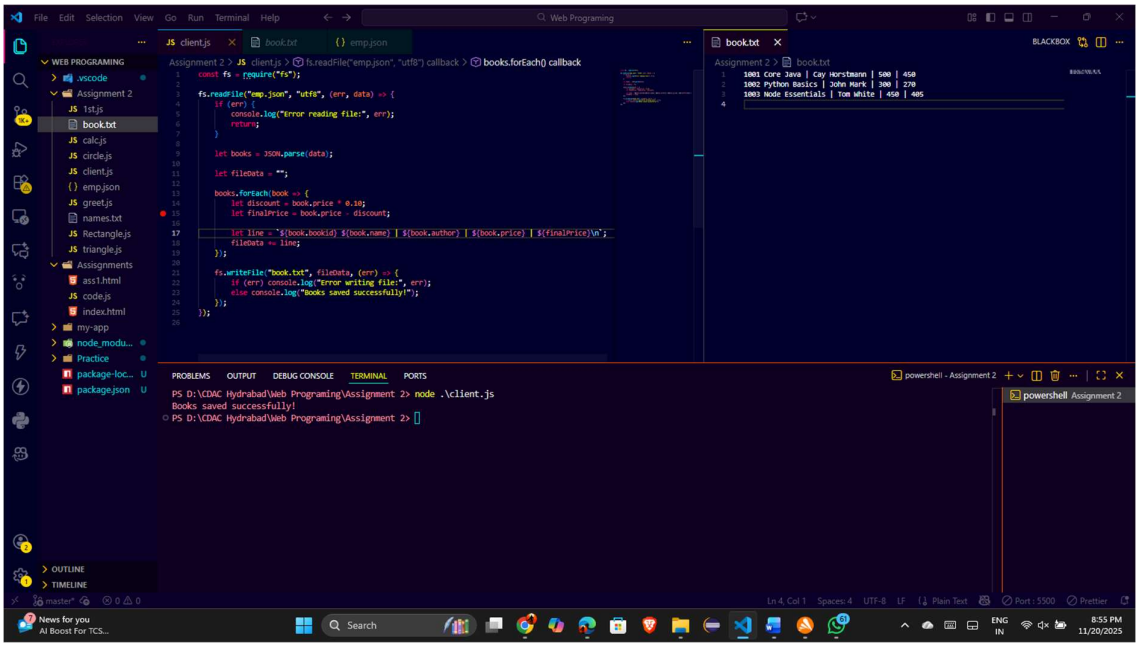
Q9)


const fs = require("fs");
let employees = [
{ empid: 1001, empname: "Yash", dept: "Sales", salary: 99000 },
{ empid: 1002, empname: "Harsh", dept: "Accounts", salary: 220000 },
{ empid: 1003, empname: "Ram", dept: "TechSupport", salary: 45000 }
];
let fileData = "";
for (let emp of employees) {
let line = `${emp.empid}:${emp.empname}:${emp.dept}:${emp.salary}\n`;
fileData += line;
}
fs.writeFile("emp.txt", fileData, (err) => {
if (err) {
console.log("Error writing file:", err);
} else {
console.log("Employee data saved successfully!");
}
});
1001:Yash:Sales:99000
1002:Harsh:Accounts:220000
1003:Ram:TechSupport:45000
PS D:\CDAC Hydrabad\Web Programing\Assignment 2> node .\client.js
Total Salary = 364000
PS D:\CDAC Hydrabad\Web Programing\Assignment 2> node .\client.js
Employee data saved successfully!
PS D:\CDAC Hydrabad\Web Programing\Assignment 2>

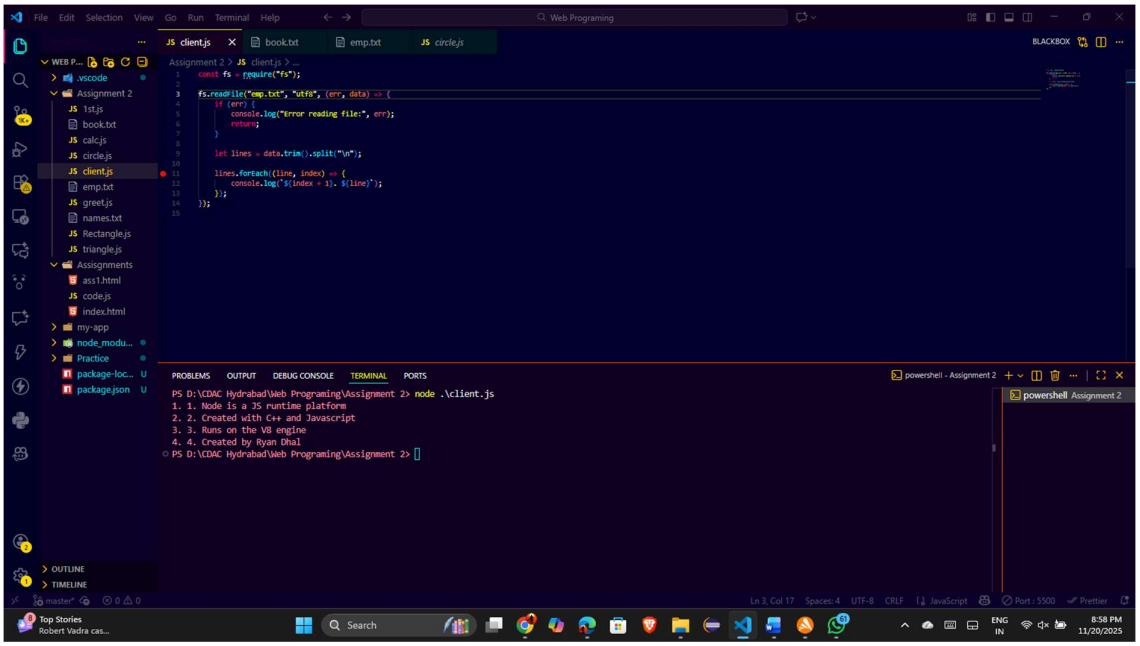
Q9)


client.js
emp.json
const fs = require("fs");
// Read the JSON file
fs.readFile("emp.json", "utf8", (err, data) => {
if (err) {
console.log("Error reading file:", err);
return;
}
// Parse JSON
let customers = JSON.parse(data);
// Display customer details
customers.forEach(customer => {
console.log("Customer Name:", customer.custname);
console.log("Address:", customer.address);
console.log("Phone No:", customer.phno);
console.log("Rating:", customer.rating);
console.log("--------------------------");
});
});
[
{
"custname": "Anil Patil",
"address": "abd 123",
"phno": "8877669988",
"rating": 8
},
{
"custname": "Anita Kulkarni",
"address": "A102, Highstreet",
"phno": "99675456",
"rating": 7
},
{
"custname": "Kavita Menon",
"address": "88203, Pune",
"phno": "123456789",
"rating": 9
}
]
PS D:\CDAC Hydrabad\Web Programing\Assignment 2> node .\client.js
PS D:\CDAC Hydrabad\Web Programing\Assignment 2> node .\client.js
Customer Name: Anil Patil
Address: abd 123
Phone No: 8877669988
Rating: 8
Customer Name: Anita Kulkarni
Address: A102, Highstreet
Phone No: 99675456
Rating: 7
Customer Name: Kavita Menon
Address: 88203, Pune
Phone No: 123456789
Rating: 9

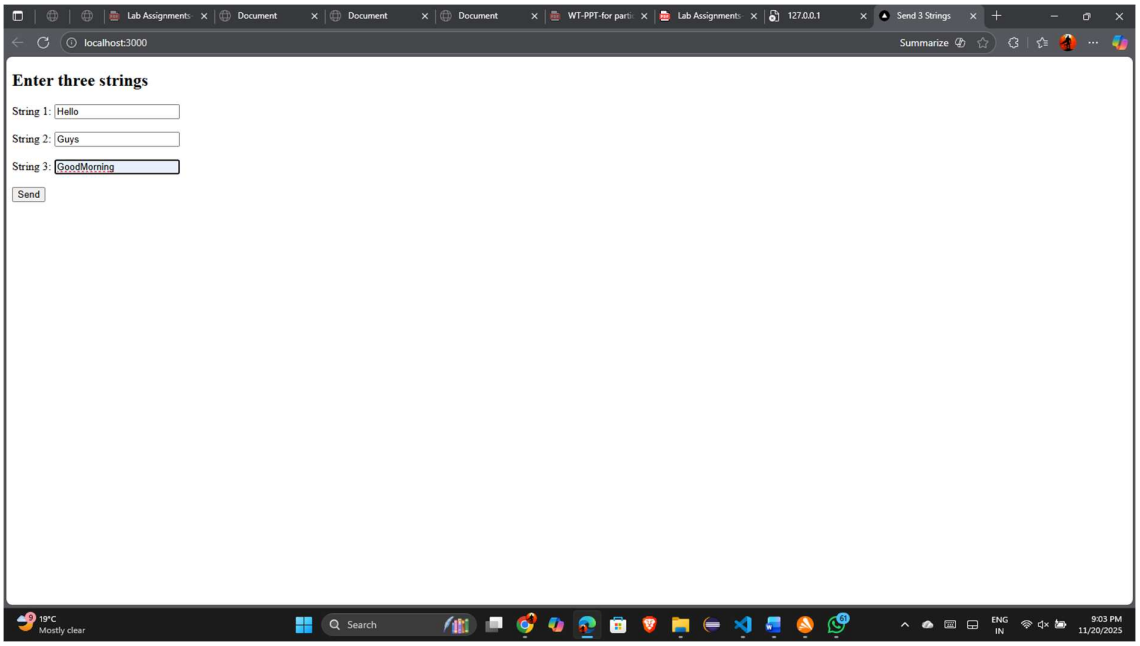
Q10)

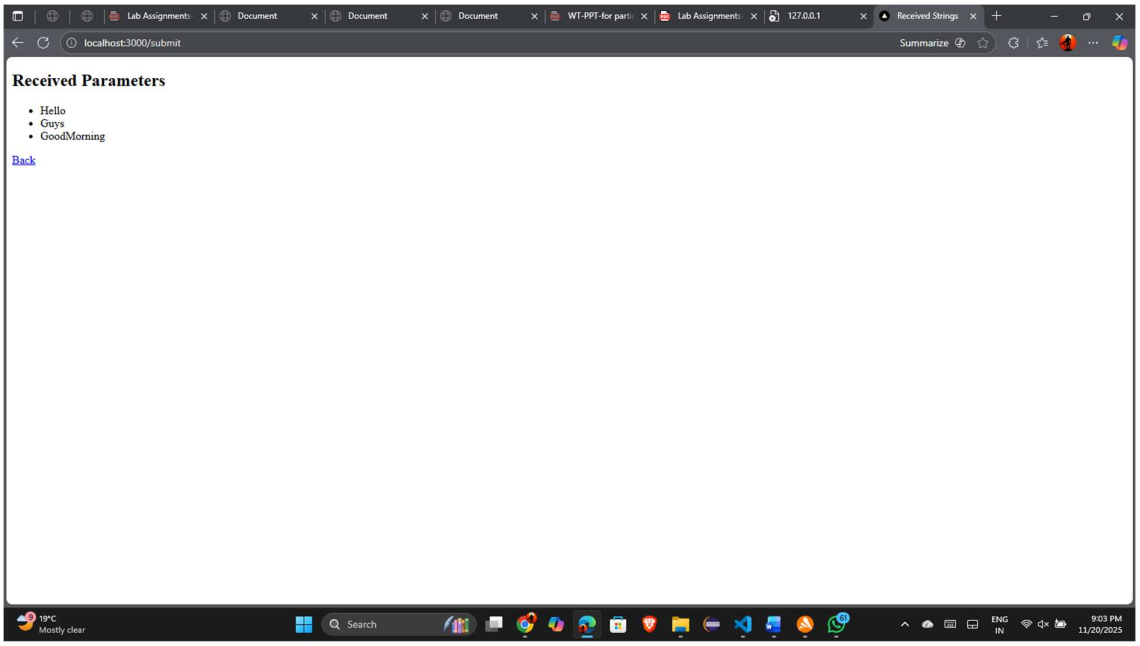

client.js
book.txt
const fs = require("fs");
fs.readFile("emp.json", "utf8", (err, data) => {
if (err) {
console.log("Error reading file:", err);
return;
}
let books = JSON.parse(data);
let fileData = "";
books.forEach(book => {
let discount = book.price * 0.10;
let finalPrice = book.price - discount;
let line = `${book.bookid} ${book.name} | ${book.author} | ${book.price} | ${finalPrice}\n`;
fileData += line;
});
fs.writeFile("book.txt", fileData, (err) => {
if (err) console.log("Error writing file:", err);
else console.log("Books saved successfully!");
});
});
1001 Core Java | Cay Horstmann | 500 | 450
1002 Python Basics | John Mark | 300 | 270
1003 Node Essentials | Tom White | 450 | 405
PS D:\CDAC Hydrabad\Web Programing\Assignment 2> node .\client.js
Books saved successfully!
PS D:\CDAC Hydrabad\Web Programing\Assignment 2>

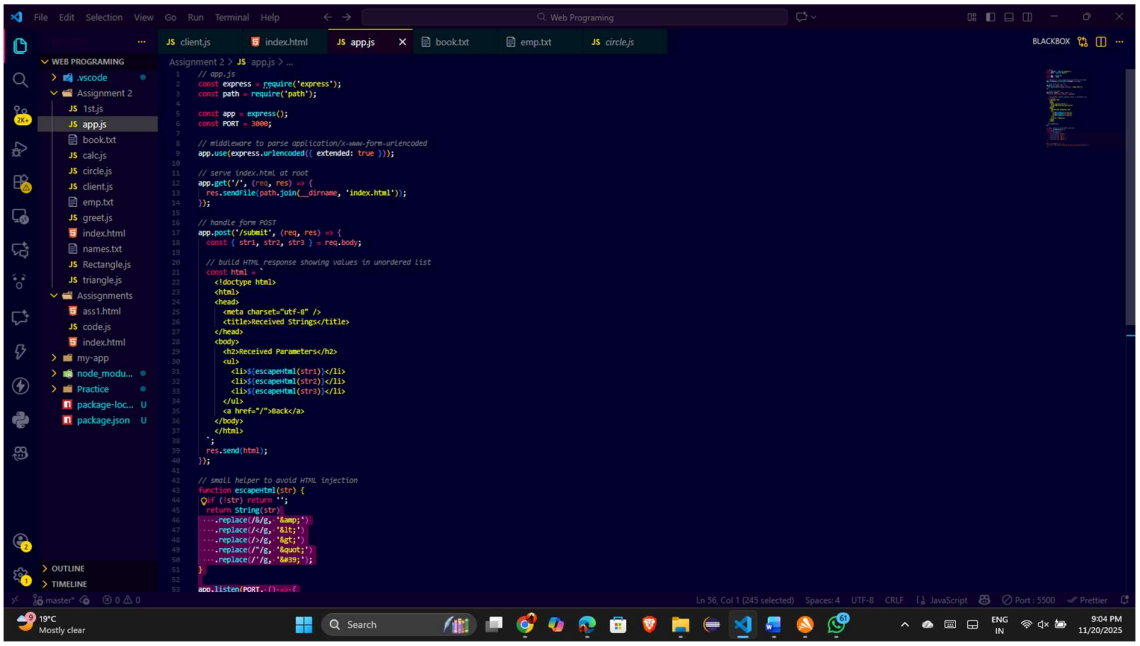
Q11)


client.js
book.txt
emp.txt
circle.js
const fs = require("fs");
fs.readFile("emp.txt", "utf8", (err, data) => {
if (err) {
console.log("Error reading file:", err);
return;
}
let lines = data.trim().split("\n");
lines.forEach((line, index) => {
console.log(`${index + 1}. ${line}`);
});
});
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
PS D:\CDAC Hydrabad\Web Programing\Assignment 2> node .\client.js
1. 1. Node is a JS runtime platform
2. 2. Created with C++ and Javascript
3. 3. Runs on the V8 engine
4. 4. Created by Ryan Dhal
PS D:\CDAC Hydrabad\Web Programing\Assignment 2>

Q11)


localhost:3000
Enter three strings
String 1: Hello
String 2: Guys
String 3: GoodMorning
Send


Lab Assignments
Document
Document
Document
WT-PPT-for partic
Lab Assignments
127.0.0.1
Received Strings
localhost:3000/submit
Summarize
Received Parameters
Hello
Guys
GoodMorning
Back
19°C
Mostly clear
Search
ENG
IN
9:03 PM
11/20/2025


File Edit Selection View Go Run Terminal Help
Web Programing
client.js
index.html
app.js
book.txt
emp.txt
circle.js
BLACKBOX
WEB PROGRAMING
.vscode
Assignment 2
1st.js
app.js
book.txt
calc.js
circle.js
client.js
emp.txt
greet.js
index.html
names.txt
Rectangle.js
triangle.js
Assisgnments
ass1.html
code.js
index.html
my-app
node_modu...
Practice
package-loc... U
package.json U
OUTLINE
TIMELINE
Assignment 2 > app.js > ...
// app.js
const express = require('express');
const path = require('path');
const app = express();
const PORT = 3000;
// middleware to parse application/x-www-form-urlencoded
app.use(express.urlencoded({ extended: true }));
// serve index.html at root
app.get('/', (req, res) => {
res.sendFile(path.join(__dirname, 'index.html'));
});
// handle form POST
app.post('/submit', (req, res) => {
const { str1, str2, str3 } = req.body;
// build HTML response showing values in unordered list
const html = `
<!doctype html>
<html>
<head>
<meta charset="utf-8" />
<title>Received Strings</title>
</head>
<body>
<h2>Received Parameters</h2>

${escapeHtml(str1)}
${escapeHtml(str2)}
${escapeHtml(str3)}

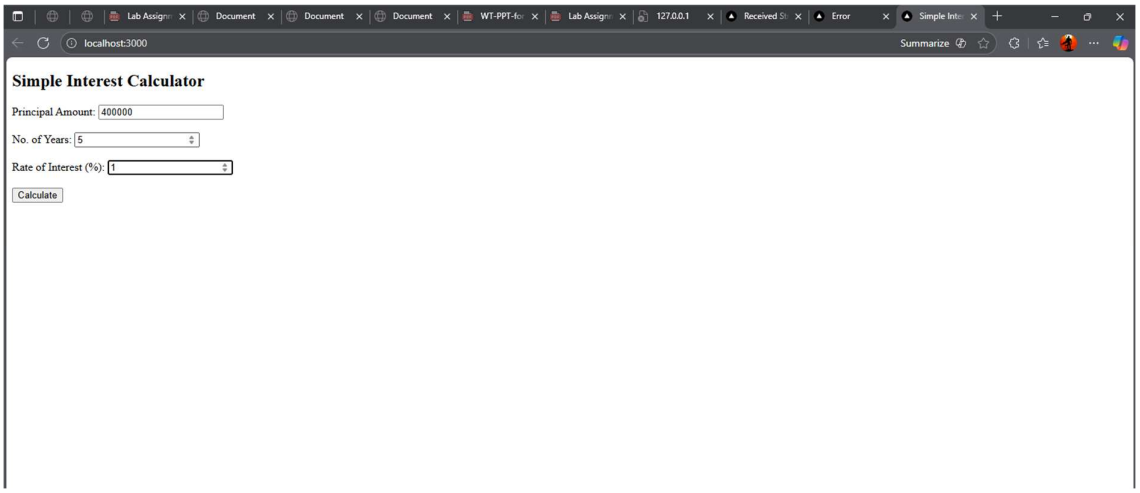
Back
</body>
</html>
`;
res.send(html);
});
// small helper to avoid HTML injection
function escapeHtml(str) {
if (!str) return '';
return String(str)
.replace(/&/g, '&')
.replace(/</g, '<')
.replace(/>/g, '>')
.replace(/"/g, '"')
.replace(/'/g, ''');
}
app.listen(PORT, () => {
Ln 56, Col 1 (245 selected) Spaces: 4 UTF-8 CRLF JavaScript Port : 5500 Prettier
19°C
Mostly clear
Search
ENG
IN
9:04 PM
11/20/2025

client.js | index.html Assignment 2 | app.js | index.html Assignments | book.txt | emp.txt

Assignment 2 > index.html > ...

```
<!doctype html>
<html>
<head>
  <meta charset="utf-8" />
  <title>Send 3 Strings</title>
</head>
<body>
  <h2>Enter three strings</h2>
  <form action="/submit" method="post">
    <label>String 1: <input type="text" name="str1" required></label><br><br>
    <label>String 2: <input type="text" name="str2" required></label><br><br>
    <label>String 3: <input type="text" name="str3" required></label><br><br>
    <button type="submit">Send</button>
  </form>
</body>
</html>
```

Q12)



\

JS client.js | index.html Assignment 2 | JS app.js | index.html Assisgnments | book.txt

Assignment 2 > JS app.js > app.get('/') callback

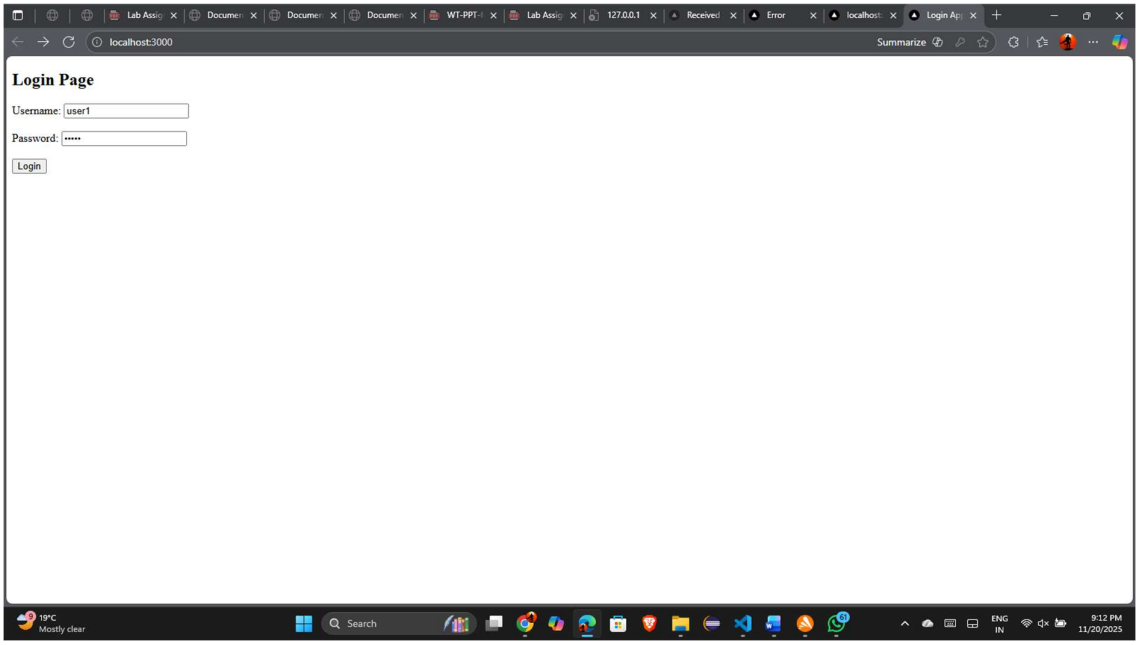
```js
const express = require('express');
const path = require('path');

const app = express();
const PORT = 3000;

// Serve HTML file
app.get('/', (req, res) => {
    res.sendFile(path.join(__dirname, 'index.html'));
});

// Calculate Simple Interest
app.get('/calculate', (req, res) => {
    const p = parseFloat(req.query.p);
    const t = parseFloat(req.query.t);
    const r = parseFloat(req.query.r);

    const si = (p * t * r) / 100;

    res.send(`
        <h2>Simple Interest Result</h2>
        <p>Principal: ${p}</p>
        <p>Years: ${t}</p>
        <p>Rate: ${r}%</p>
        <h3>Simple Interest = ${si}</h3>
        <a href="/">Go Back</a>
    `);
});

app.listen(PORT, () => {
    console.log(`Server running at http://localhost:${PORT}`);
});
```

JS client.js | index.html Assignment 2 | JS app.js | index.html Assisgnments

Assignment 2 > index.html > ...

```html
<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <title>Simple Interest Calculator</title>
</head>
<body>
    <h2>Simple Interest Calculator</h2>

    <form action="/calculate" method="get">
        <label>Principal Amount: </label>
        <input type="number" name="p" required><br><br>

        <label>No. of Years: </label>
        <input type="number" name="t" required><br><br>

        <label>Rate of Interest (%): </label>
        <input type="number" name="r" required><br><br>

        <button type="submit">Calculate</button>
    </form>
</body>
</html>
```

Q13)



Welcome, user1! Login Successful ✓

Logout

JS client.js | index.html Assignment 2 | JS app.js | index.html Assignments | book.txt | emp.txt

Assignment 2 > JS app.js > app.get('/') callback

```js
const express = require('express');
const path = require('path');
const app = express();

const PORT = 3000;

// User database (array)
let users = [
    { uname: "shrilata", pass: "secret" },
    { uname: "admin", pass: "admin123" },
    { uname: "user1", pass: "pass1" }
];

// Serve login page
app.get('/', (req, res) => {
    res.sendFile(path.join(__dirname, 'index.html'));
});

// Handle login request
app.get('/login', (req, res) => {
    const uname = req.query.uname;
    const pass = req.query.pass;

    // Check if user exists
    const found = users.find(u => u.uname === uname && u.pass === pass);

    if (found) {
        res.send(`<h2>Welcome, ${uname}! Login Successful ✓</h2>
                  <a href="/">Logout</a>`);
    } else {
        res.send(`<h2 style="color:red;">Invalid Username or Password ✗</h2>
                  <a href="/">Try Again</a>`);
    }
});

// Start server
app.listen(PORT, () => {
    console.log(`Server running at http://localhost:${PORT}`);
});

```

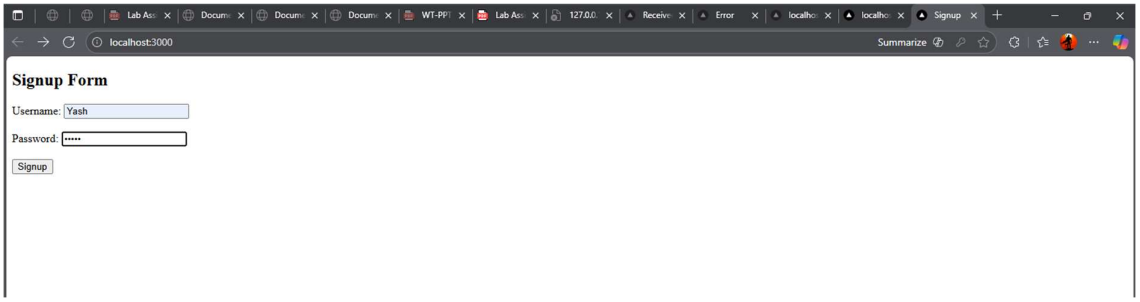
JS client.js | index.html Assignment 2 | JS app.js | index.html Assignments | book.txt

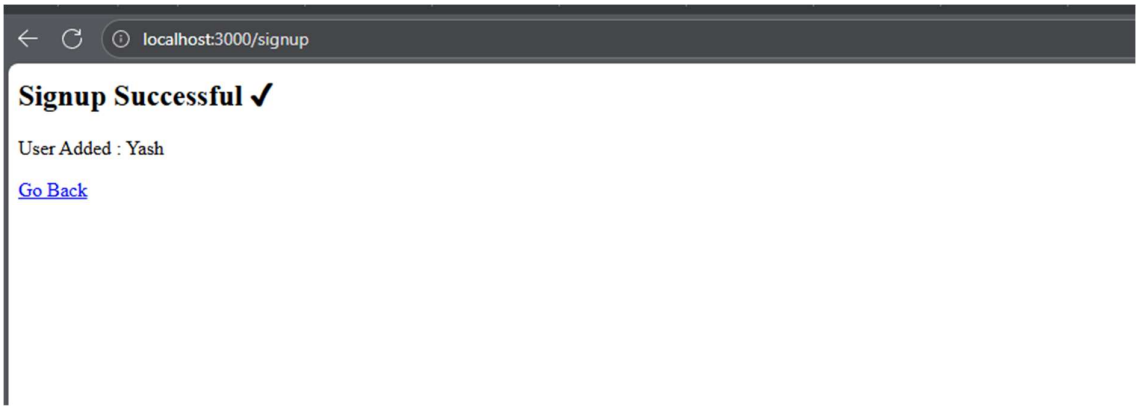
Assignment 2 > index.html > html > body > form

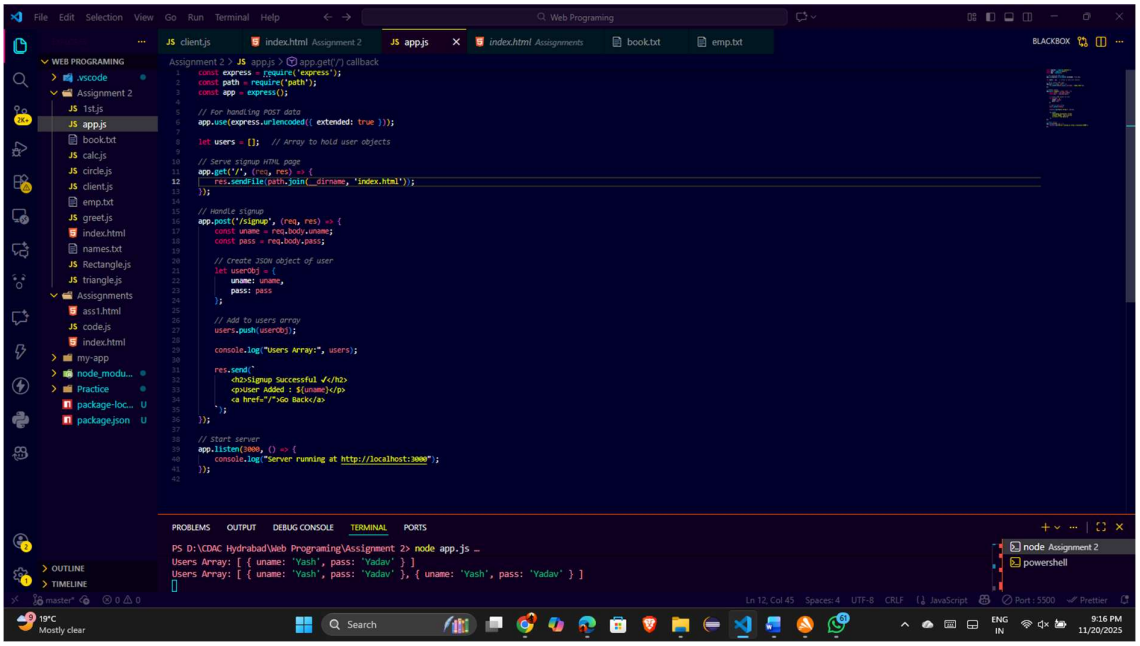
```html
<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <title>Login App</title>
</head>
<body>
    <h2>Login Page</h2>

    <form action="/login" method="get">
        <label>Username: </label>
        <input type="text" name="uname" required><br><br>

        <label>Password: </label>
        <input type="password" name="pass" required><br><br>

        <button type="submit">Login</button>
    </form>
</body>
</html>

```

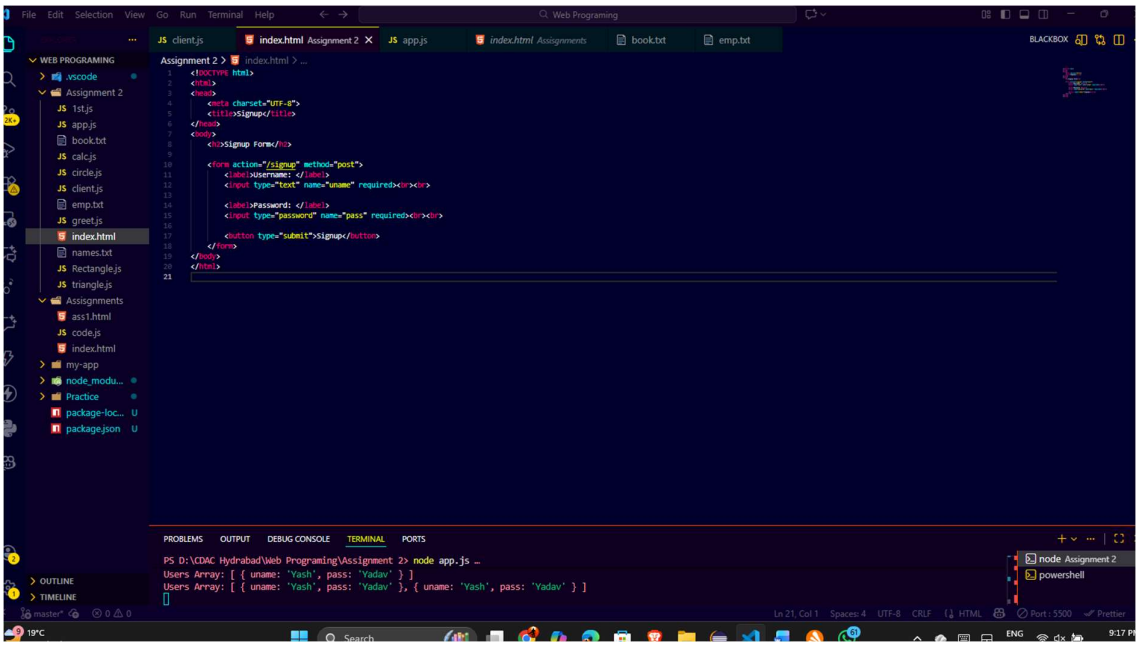
Chat (CTRL + I) / Share (CTRL + L)

Q14)


localhost:3000
Summarize
Signup Form
Username: Yash
Password:
Signup


localhost:3000/signup
Signup Successful ✓
User Added : Yash
Go Back

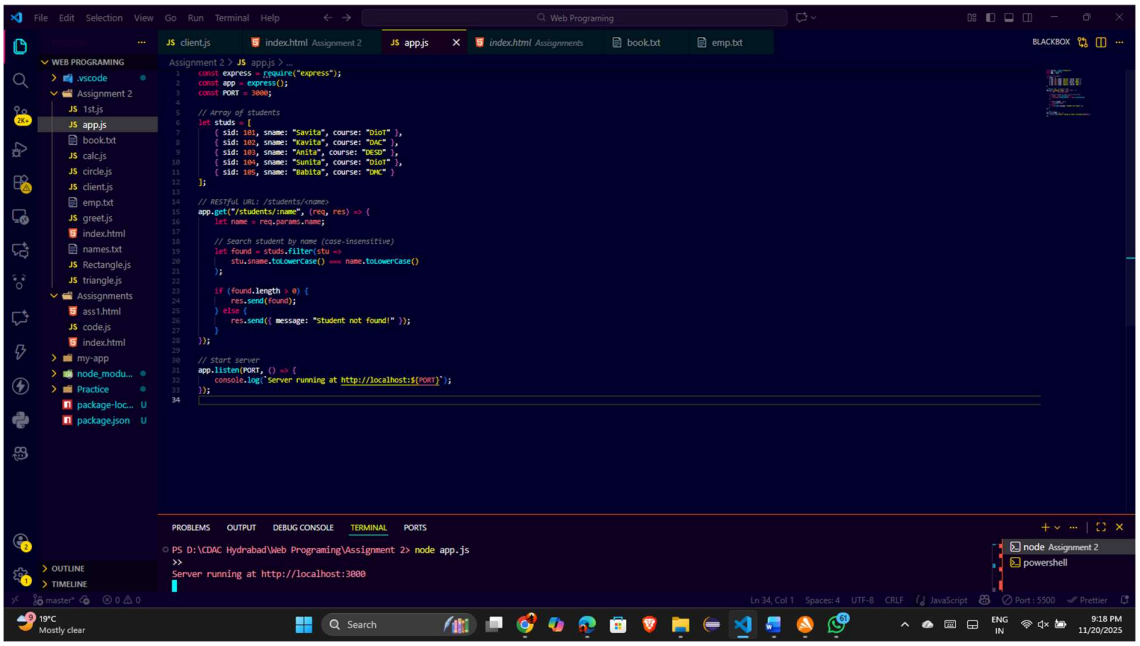

File Edit Selection View Go Run Terminal Help
Web Programing
client.js
index.html Assignment 2
app.js
index.html Assignments
book.txt
emp.txt
BLACKBOX
WEB PROGRAMING
.vscode
Assignment 2
1st.js
app.js
book.txt
calc.js
circle.js
client.js
emp.txt
greet.js
index.html
names.txt
Rectangle.js
triangle.js
Assisgnments
ass1.html
code.js
index.html
my-app
node_modu...
Practice
package-loc... U
package.json U
OUTLINE
TIMELINE
const express = require('express');
const path = require('path');
const app = express();
// For handling POST data
app.use(express.urlencoded({ extended: true }));
let users = []; // Array to hold user objects
// Serve signup HTML page
app.get('/', (req, res) => {
res.sendFile(path.join(__dirname, 'index.html'));
});
// Handle signup
app.post('/signup', (req, res) => {
const uname = req.body.uname;
const pass = req.body.pass;
// Create JSON object of user
let userObj = {
uname: uname,
pass: pass
};
// Add to users array
users.push(userObj);
console.log("Users Array:", users);
res.send(`
<h2>Signup Successful ✓</h2>
User Added : ${uname}
Go Back
`);
});
// Start server
app.listen(3000, () => {
console.log("Server running at http://localhost:3000");
});
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
PS D:\CDAC Hydrabad\Web Programing\Assignment 2> node app.js
Users Array: [{ uname: 'Yash', pass: 'Yadav' }]
Users Array: [{ uname: 'Yash', pass: 'Yadav' }, { uname: 'Yash', pass: 'Yadav' }]
node Assignment 2
powershell
19°C
Mostly clear
Search
ENG
IN
9:16 PM
11/20/2025

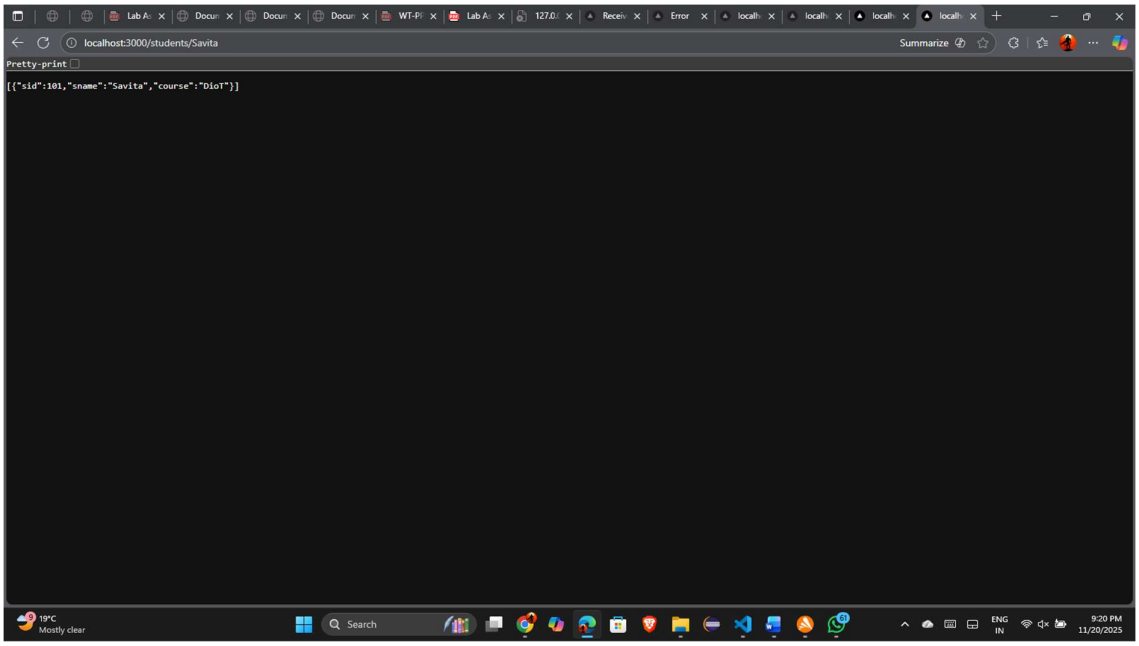

File Edit Selection View Go Run Terminal Help
Web Programing
client.js
index.html Assignment 2
app.js
index.html Assignments
book.txt
emp.txt
BLACKBOX
WEB PROGRAMING
.vscode
Assignment 2
1st.js
app.js
book.txt
calc.js
circle.js
client.js
emp.txt
greet.js
index.html
names.txt
Rectangle.js
triangle.js
Assisgnments
ass1.html
code.js
index.html
my-app
node_modu...
Practice
package-loc... U
package.json U
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Signup</title>
</head>
<body>
<h1>Signup Form</h1>
<form action="/signup" method="post">
<label>Username: </label>
<input type="text" name="uname" required>

<label>Password: </label>
<input type="password" name="pass" required>

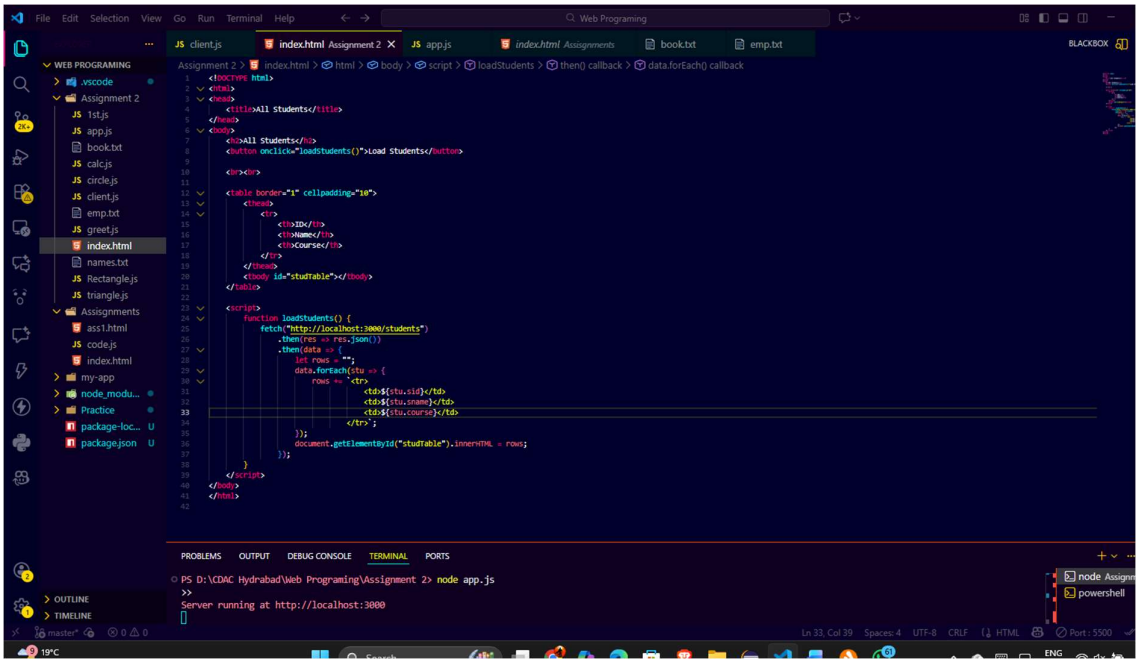
<button type="submit">Signup</button>
</form>
</body>
</html>
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
PS D:\CDAC Hydrabad\Web Programing\Assignment 2> node app.js …
Users Array: [{ uname: 'Yash', pass: 'Yadav' }]
Users Array: [{ uname: 'Yash', pass: 'Yadav' }, { uname: 'Yash', pass: 'Yadav' }]
node Assignment 2
powershell
OUTLINE
TIMELINE
19°C
Search
ENG
9:17 PM

Q15)


File Edit Selection View Go Run Terminal Help
Web Programing
client.js
index.html Assignment 2
app.js
index.html Assignments
book.txt
emp.txt
BLACKBOX
WEB PROGRAMING
.vscode
Assignment 2
1st.js
app.js
book.txt
calc.js
circle.js
client.js
emp.txt
greet.js
index.html
names.txt
Rectangle.js
triangle.js
Assisgnments
ass1.html
code.js
index.html
my-app
node_modu...
Practice
package-loc... U
package.json U
const express = require("express");
const app = express();
const PORT = 3000;
// Array of students
let studs = [
{ sid: 101, sname: "Savita", course: "DIoT" },
{ sid: 102, sname: "Kavita", course: "DAC" },
{ sid: 103, sname: "Anita", course: "DESD" },
{ sid: 104, sname: "Sunita", course: "DIoT" },
{ sid: 105, sname: "Babita", course: "DMC" }
];
// RESTful URL: /students/<name>
app.get("/students/:name", (req, res) => {
let name = req.params.name;
// Search student by name (case-insensitive)
let found = studs.filter(stu =>
stu.sname.toLowerCase() === name.toLowerCase()
);
if (found.length > 0) {
res.send(found);
} else {
res.send({ message: "Student not found!" });
}
});
// Start server
app.listen(PORT, () => {
console.log(`Server running at http://localhost:${PORT}`);
});
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
PS D:\CDAC Hydrabad\Web Programing\Assignment 2> node app.js
>>
Server running at http://localhost:3000
node Assignment 2
powershell
OUTLINE
TIMELINE
19°C
Mostly clear
Search
ENG
IN
9:18 PM
11/20/2025

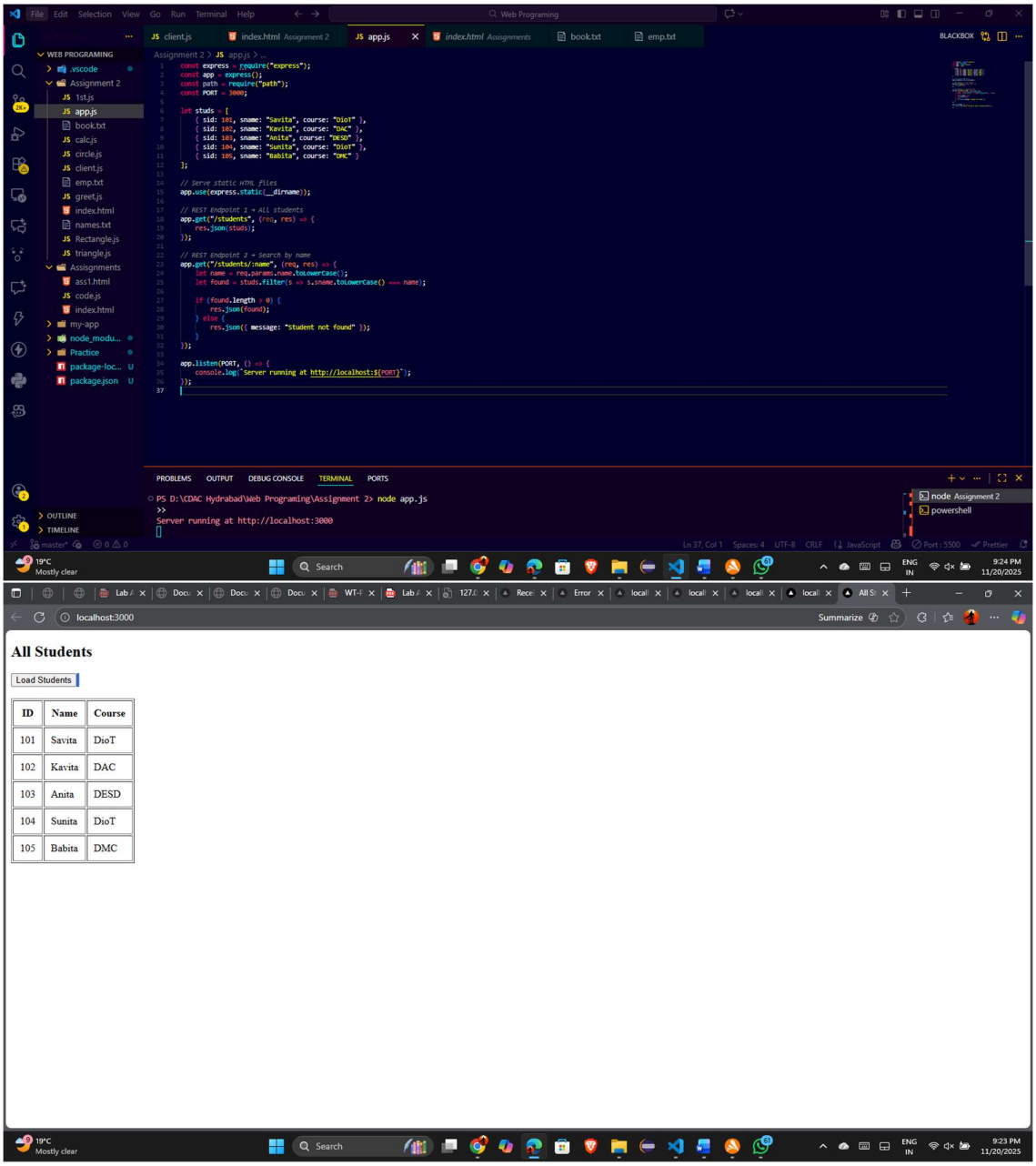

localhost:3000/students/Savita
Pretty-print
[{"sid":101,"sname":"Savita","course":"DioT"}]

Q16)


index.html
<!DOCTYPE html>
<html>
<head>
<title>All Students</title>
</head>
<body>
<h2>All Students</h2>
<button onclick="loadStudents()">Load Students</button>

<table border="1" cellpadding="10">
<thead>
<tr>
<th>ID</th>
<th>Name</th>
<th>Course</th>
</tr>
</thead>
<tbody id="studTable"></tbody>
</table>
<script>
function loadStudents() {
fetch("http://localhost:3000/students")
.then(res => res.json())
.then(data => {
let rows = "";
data.forEach(stu => {
rows += `<tr>
<td>${stu.sid}</td>
<td>${stu.sname}</td>
<td>${stu.course}</td>
</tr>`;
});
document.getElementById("studTable").innerHTML = rows;
});
}
</script>
</body>
</html>
PS D:\CDAC Hydrabad\Web Programing\Assignment 2> node app.js
Server running at http://localhost:3000

1.


File Edit Selection View Go Run Terminal Help
Web Programing
WEB PROGRAMING
.vscode
Assignment 2
1st.js
app.js
book.txt
calc.js
circle.js
client.js
emp.txt
greet.js
index.html
names.txt
Rectangle.js
triangle.js
Assisgnments
ass1.html
code.js
index.html
my-app
node_modu...
Practice
package-loc...
package.json
OUTLINE
TIMELINE
client.js
index.html Assignment 2
app.js
index.html Assignments
book.txt
emp.txt
BLACKBOX
Assignment 2 > app.js > ...
const express = require("express");
const app = express();
const path = require("path");
const PORT = 3000;
let studs = [
{ sid: 101, sname: "Savita", course: "DioT" },
{ sid: 102, sname: "Kavita", course: "DAC" },
{ sid: 103, sname: "Anita", course: "DESD" },
{ sid: 104, sname: "Sunita", course: "DioT" },
{ sid: 105, sname: "Babita", course: "DMC" }
];
// Serve static HTML files
app.use(express.static(__dirname));
// REST Endpoint 1 → All students
app.get("/students", (req, res) => {
res.json(studs);
});
// REST Endpoint 2 → Search by name
app.get("/students/:name", (req, res) => {
let name = req.params.name.toLowerCase();
let found = studs.filter(s => s.sname.toLowerCase() === name);
if (found.length > 0) {
res.json(found);
} else {
res.json({ message: "Student not found" });
}
});
app.listen(PORT, () => {
console.log(`Server running at http://localhost:${PORT}`);
});
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS D:\CDAC Hydrabad\Web Programing\Assignment 2> node app.js
>>
Server running at http://localhost:3000
node Assignment 2
powershell
Ln 37, Col 1 Spaces: 4 UTF-8 CRLF JavaScript Port : 5500 Prettier
19°C
Mostly clear
Search
ENG
IN
9:24 PM
11/20/2025

localhost:3000
Summarize
All Students
Load Students
ID
Name
Course
101
Savita
DioT
102
Kavita
DAC
103
Anita
DESD
104
Sunita
DioT
105
Babita
DMC
19°C
Mostly clear
Search
ENG
IN
9:23 PM
11/20/2025