Week-6

React Applications

**Superset Id:- 4993066**

Lab-1**: Creating Your First React Application**

1. Open Command Prompt and type:

npx create-react-app myfirstreact

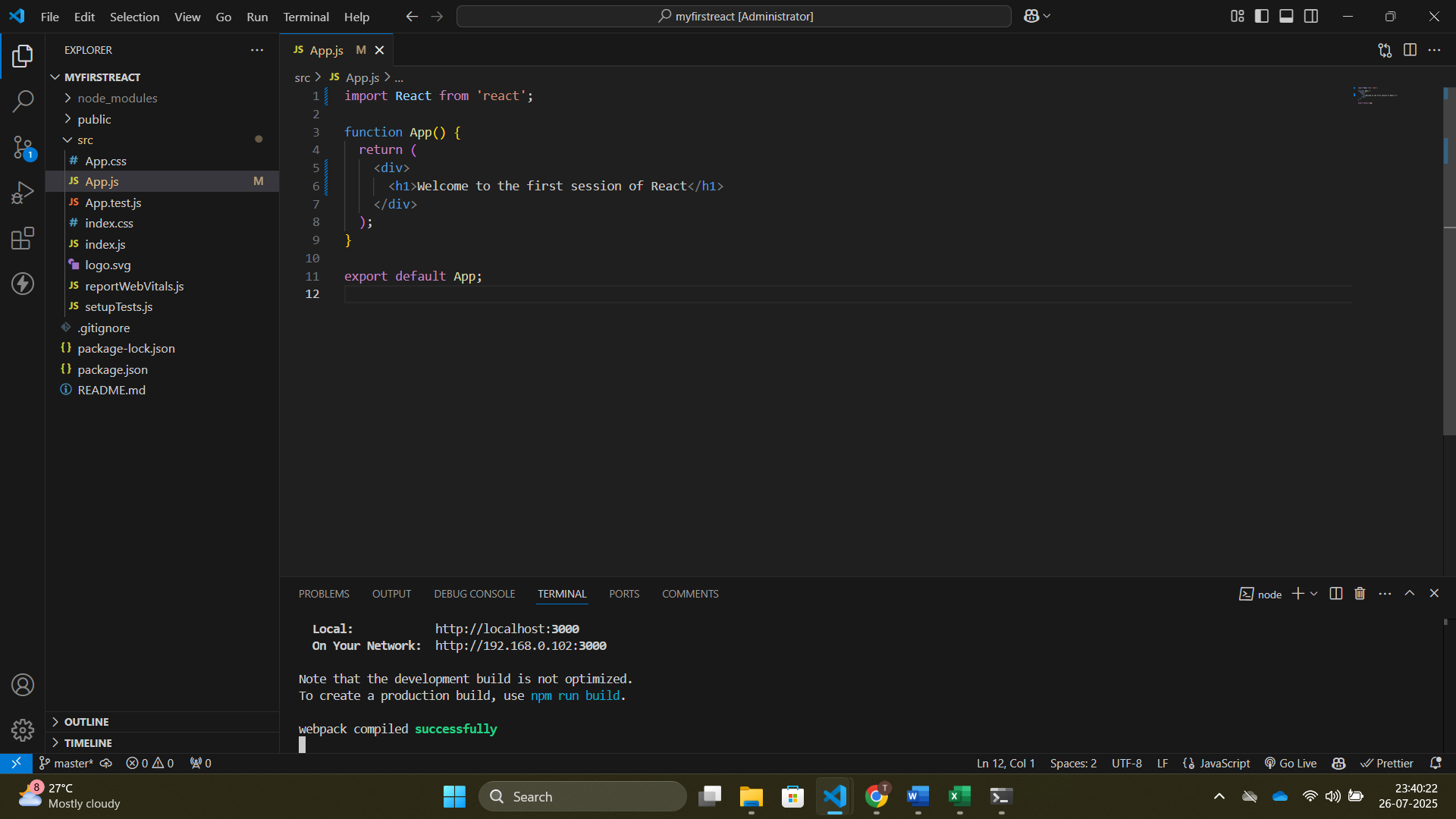
1. Navigate into the app folder:

cd myfirstreact

1. Open the folder in Visual Studio Code.

Open 'src/App.js', delete its contents, and replace you’re your custom code (e.g., 'Welcome to the first session of React').

1. npm start



A screenshot of a computer

AI-generated content may be incorrect.

Lab-2: **Creating Multiple React Components**

1. npx create-react-app StudentApp
2. Create a folder 'Components' inside 'src'.
3. Inside 'Components', create 'Home.js', 'About.js', and 'Contact.js' files.

Home.js:-

import React from 'react';

class Home extends React.Component {

  render() {

    return <h2>Welcome to the Home page of Student Management Portal</h2>;

  }

}

export default Home;

About.js:-

import React from 'react';

class About extends React.Component {

render() {

return <h2>Welcome to the About page of the Student Management Portal</h2>;

}

}

export default About;

Contact.js:-

import React from 'react';

class Contact extends React.Component {

render() {

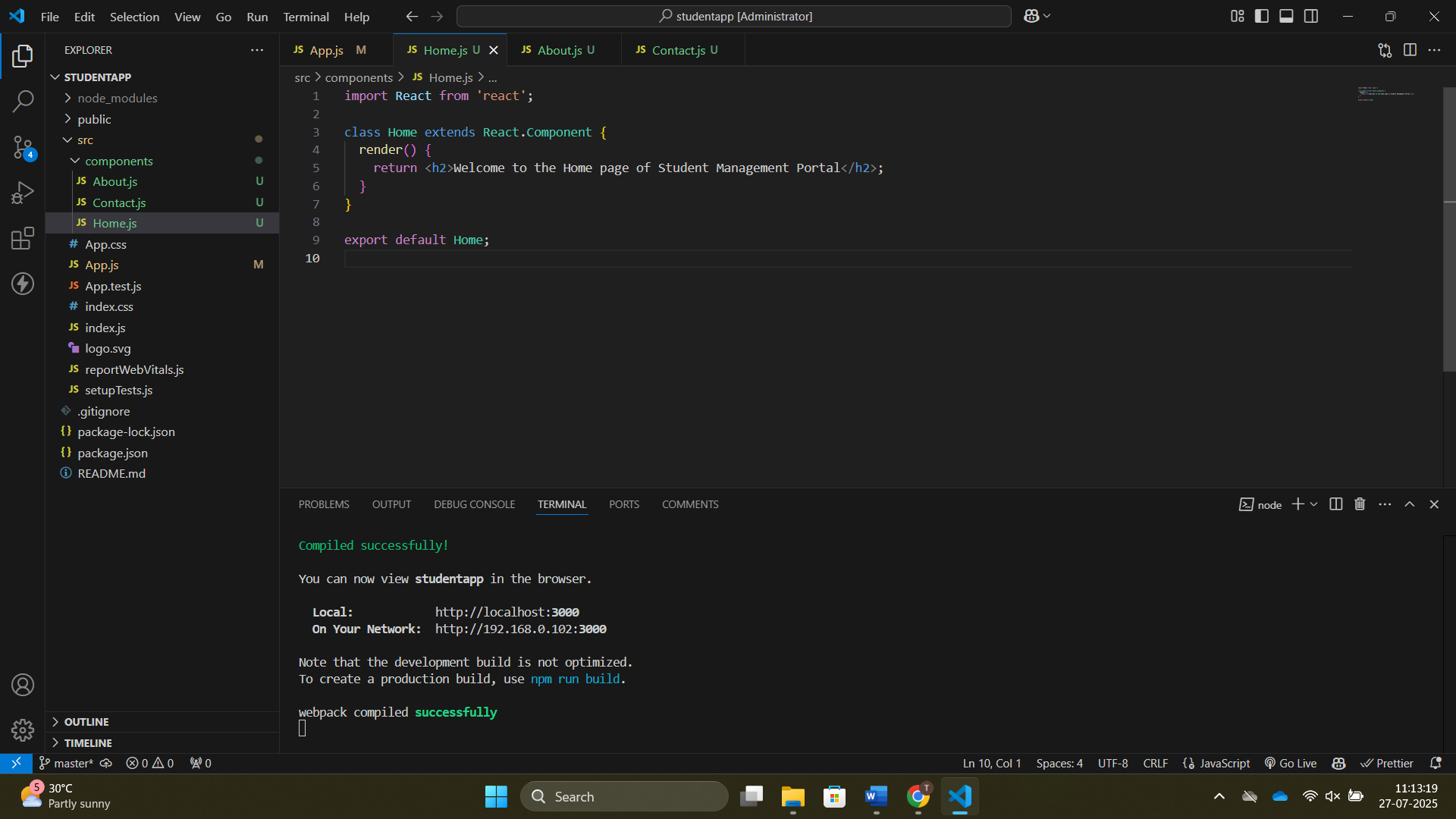
return <h2>Welcome to the Contact page of the Student Management Portal</h2>;

}

}

export default Contact;

1. npm start



A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

Lab-3: **Creating a Functional Component with Styling**

1. npx create-react-app scorecalculatorapp
2. Under 'src', create a folder named 'Components' and add 'CalculateScore.js'.
3. Create a function component to accept Name, School, Total and Goal.
4. Create a folder named 'Stylesheets' and add 'mystyle.css' for styling.
5. Edit 'App.js' to include the CalculateScore component.
6. npm start

CalculateScore.js:-

import React from 'react';

import '../stylesheets/mystyle.css'; // lowercase folder name

function CalculateScore() {

const student = {

name: "John",

school: "ABC School",

total: 450,

goal: 500

};

const average = (student.total / student.goal) \* 100;

return (

<div className="score-box">

<h2>Student Score Calculator</h2>

<p><strong>Name:</strong> {student.name}</p>

<p><strong>School:</strong> {student.school}</p>

<p><strong>Total:</strong> {student.total}</p>

<p><strong>Goal:</strong> {student.goal}</p>

<p><strong>Average Score:</strong> {average.toFixed(2)}%</p>

</div>

);

}

export default CalculateScore;

mystyle.css:-

/\* Score box main container \*/

.score-box {

    max-width: 500px;

    margin: 50px auto;

    padding: 30px;

    background: linear-gradient(to right, #e0eafc, #cfdef3);

    border-radius: 15px;

    box-shadow: 0 8px 25px rgba(0, 0, 0, 0.2);

    font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;

    color: #333;

    transition: transform 0.3s ease-in-out;

  }

  .score-box:hover {

    transform: scale(1.02);

  }

  /\* Header styling \*/

  .score-box h2 {

    text-align: center;

    font-size: 26px;

    color: #222;

    margin-bottom: 25px;

    border-bottom: 2px solid #444;

    padding-bottom: 10px;

  }

  /\* Paragraph styling \*/

  .score-box p {

    font-size: 18px;

    margin: 10px 0;

    line-height: 1.5;

  }

  .score-box strong {

    color: #555;

  }

  /\* Responsive \*/

  @media (max-width: 600px) {

    .score-box {

      width: 90%;

      padding: 20px;

    }

    .score-box h2 {

      font-size: 22px;

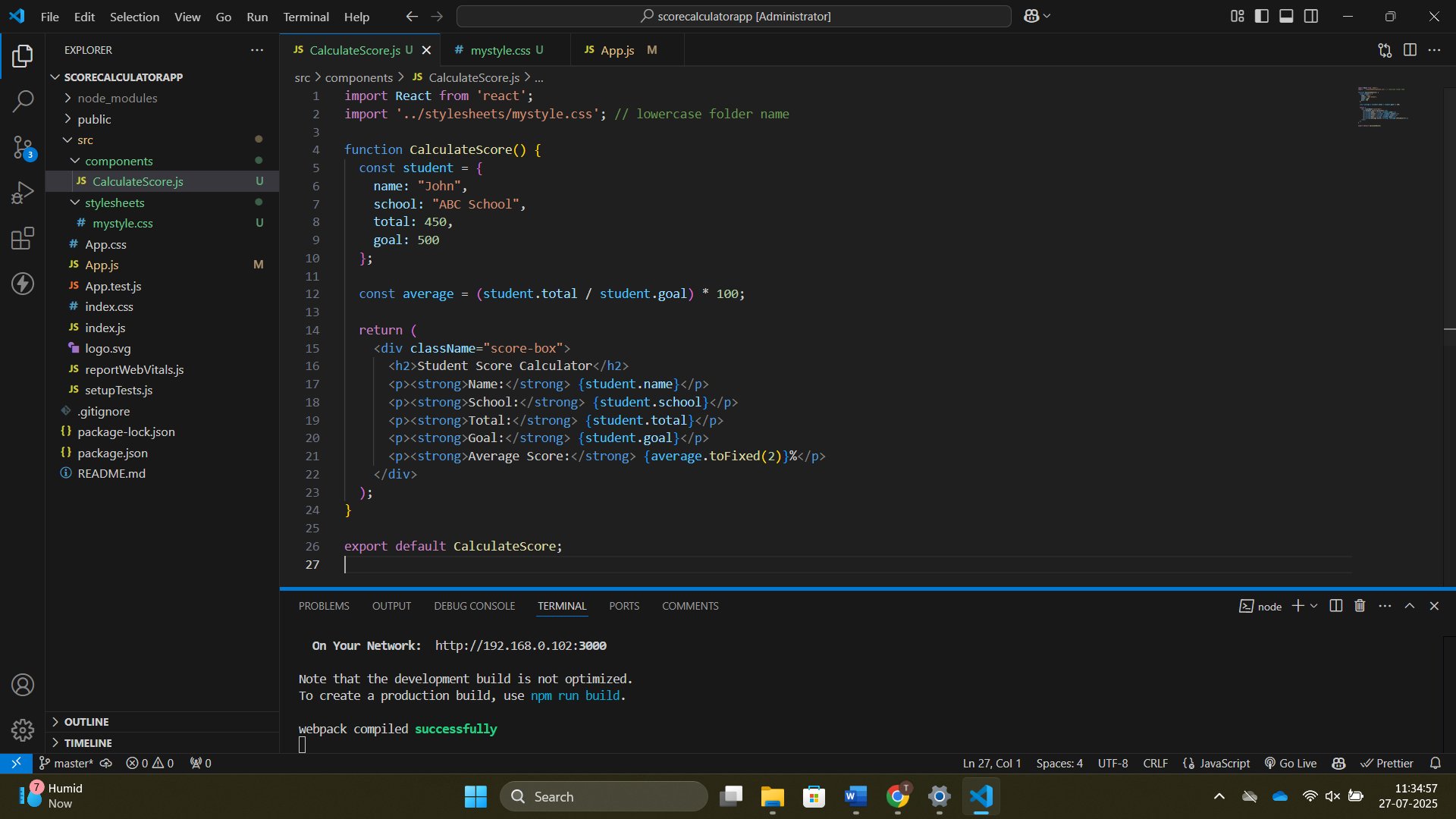
    }

    .score-box p {

      font-size: 16px;

    }

  }



A screenshot of a computer screen

AI-generated content may be incorrect.

A screenshot of a computer program

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

Lab-4:- Working with Component Lifecycle Methods

1. npx create-react-app blogapp
2. Open in VS Code and create 'Post.js' and 'Posts.js' in 'src'.
3. Edit 'App.js' to include the Posts component.
4. npm start

Post.js:

// src/Post.js

class Post {

constructor(userId, id, title, body) {

this.userId = userId;

this.id = id;

this.title = title;

this.body = body;

}

}

export default Post;

Posts.js:

// src/Posts.js

import React, { Component } from 'react';

import Post from './Post';

class Posts extends Component {

constructor(props) {

super(props);

this.state = {

posts: [

new Post(1, 101, "Who Am I?", "I am Tarun, a passionate learner and developer exploring the world of data science and web development."),

new Post(1, 102, "My Education", "Currently pursuing my degree with strong interest in programming, analytics, and AI."),

new Post(1, 103, "My Projects", "I worked on several projects including 'BidX' and a portfolio website using ReactJS."),

new Post(1, 104, "My Skills", "I am skilled in ReactJS, Python, Data Science, HTML, CSS, and teamwork."),

new Post(1, 105, "Why I Love Coding", "Because coding lets me build things from scratch and solve real-world problems."),

new Post(1, 106, "My Vision", "I aim to be a leading data scientist and full-stack developer who creates impactful solutions.")

],

hasError: false

};

}

componentDidCatch(error, info) {

alert("Something went wrong!");

console.error("Caught error:", error, info);

}

render() {

const { posts, hasError } = this.state;

if (hasError) {

return <h2>Error loading posts. Try again later.</h2>;

}

return (

<div style={{ padding: "20px" }}>

<h1>Tarun's Blog</h1>

{posts.map(post => (

<div key={post.id} style={{ marginBottom: "20px", background: "#f9f9f9", padding: "15px", borderRadius: "10px", boxShadow: "0 2px 8px rgba(0,0,0,0.1)" }}>

<h3 style={{ color: "#2c3e50" }}>{post.title}</h3>

<p style={{ color: "#555" }}>{post.body}</p>

</div>

))}

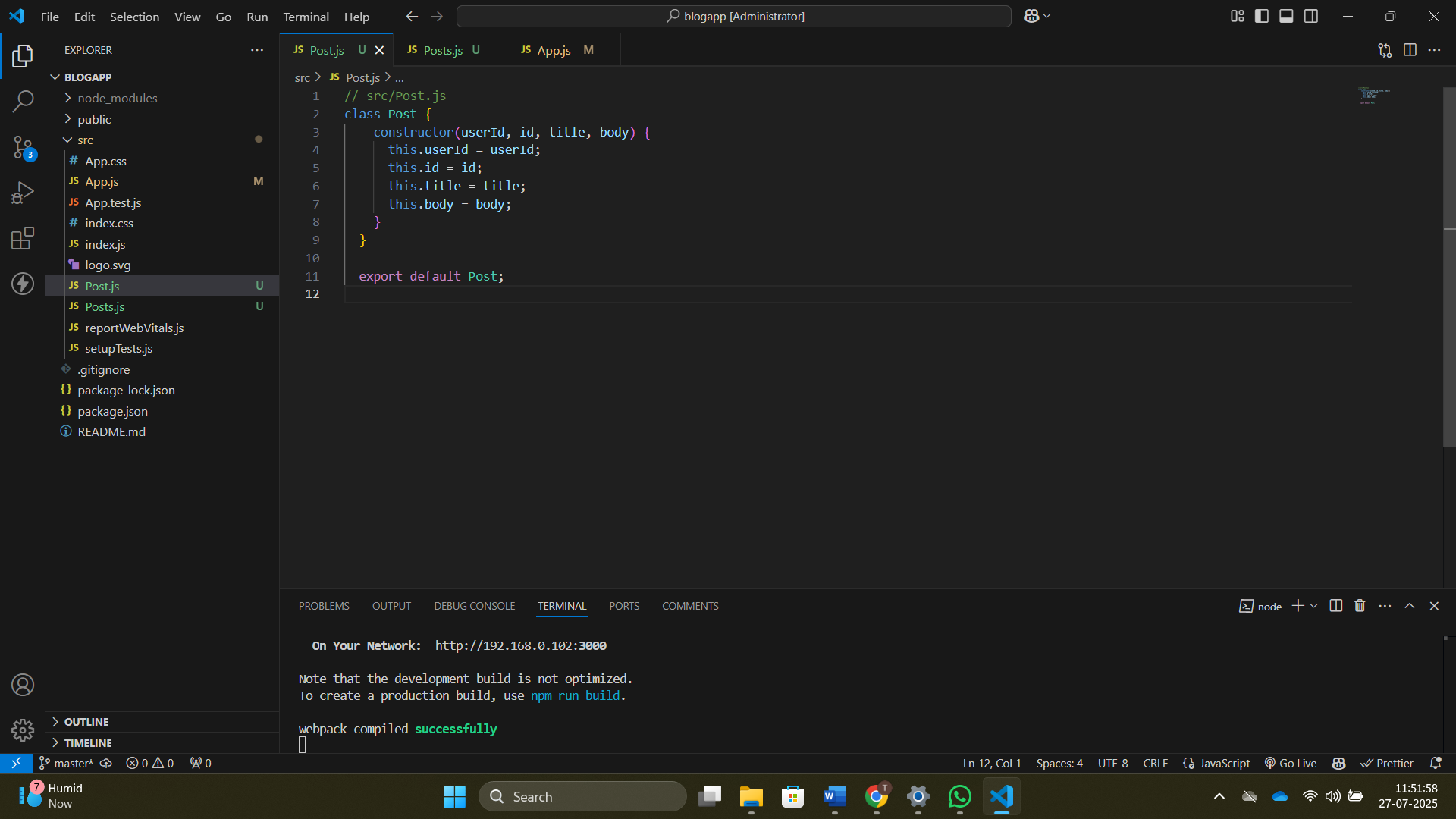
</div>

);

}

}

export default Posts;



A screenshot of a computer screen

AI-generated content may be incorrect.

A screenshot of a computer program

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

Lab-5: **Styling React Components using CSS Modules**

1. Download and unzip the provided React application folder.
2. Open Command Prompt, navigate to the folder, and restore packages:
3. npm install
4. Create a file 'CohortDetails.module.css' in 'src'.
5. Define a class 'box' with: width: 300px; display: inline-block; margin: 10px; padding: 10px 20px; border: 1px solid black; border-radius: 10px;
6. Import the CSS module in the component.
7. Apply 'box' class to container div and conditional style to <h3>.
8. npm start

cohort detailsmodule.css:-

.box {

    width: 300px;

    display: inline-block;

    margin: 10px;

    padding: 10px 20px;

    border: 1px solid black;

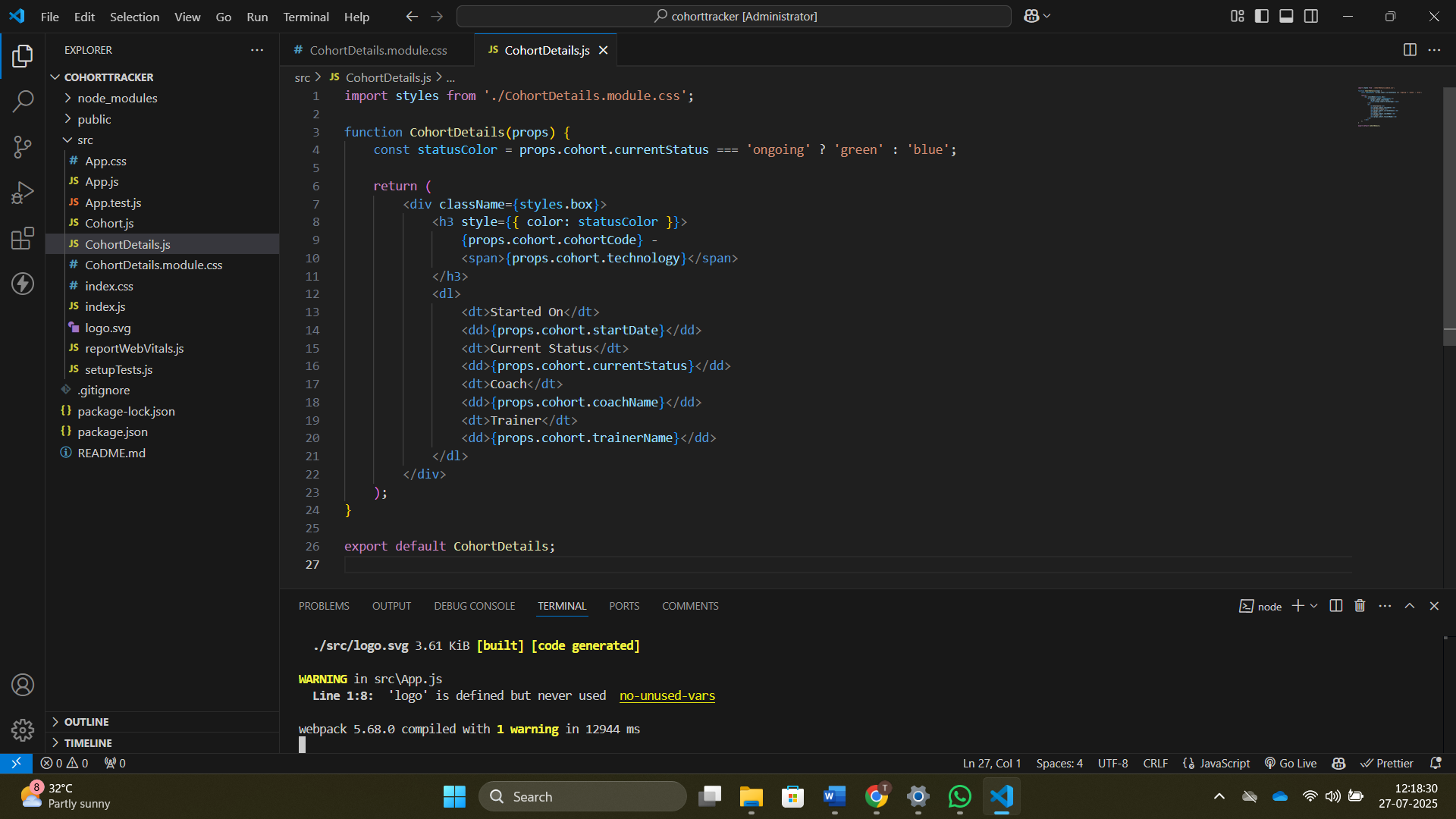
    border-radius: 10px;

  }

  dt {

    font-weight: 500;

  }



A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.