**Candidate Name**:- MEGHANA G K

**Superset ID**:-6424295

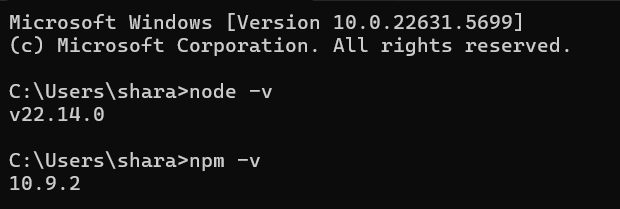
**WEEK – 6 HANDS ON EXERCISE (JAVA FSE DEEPSKILLING)**

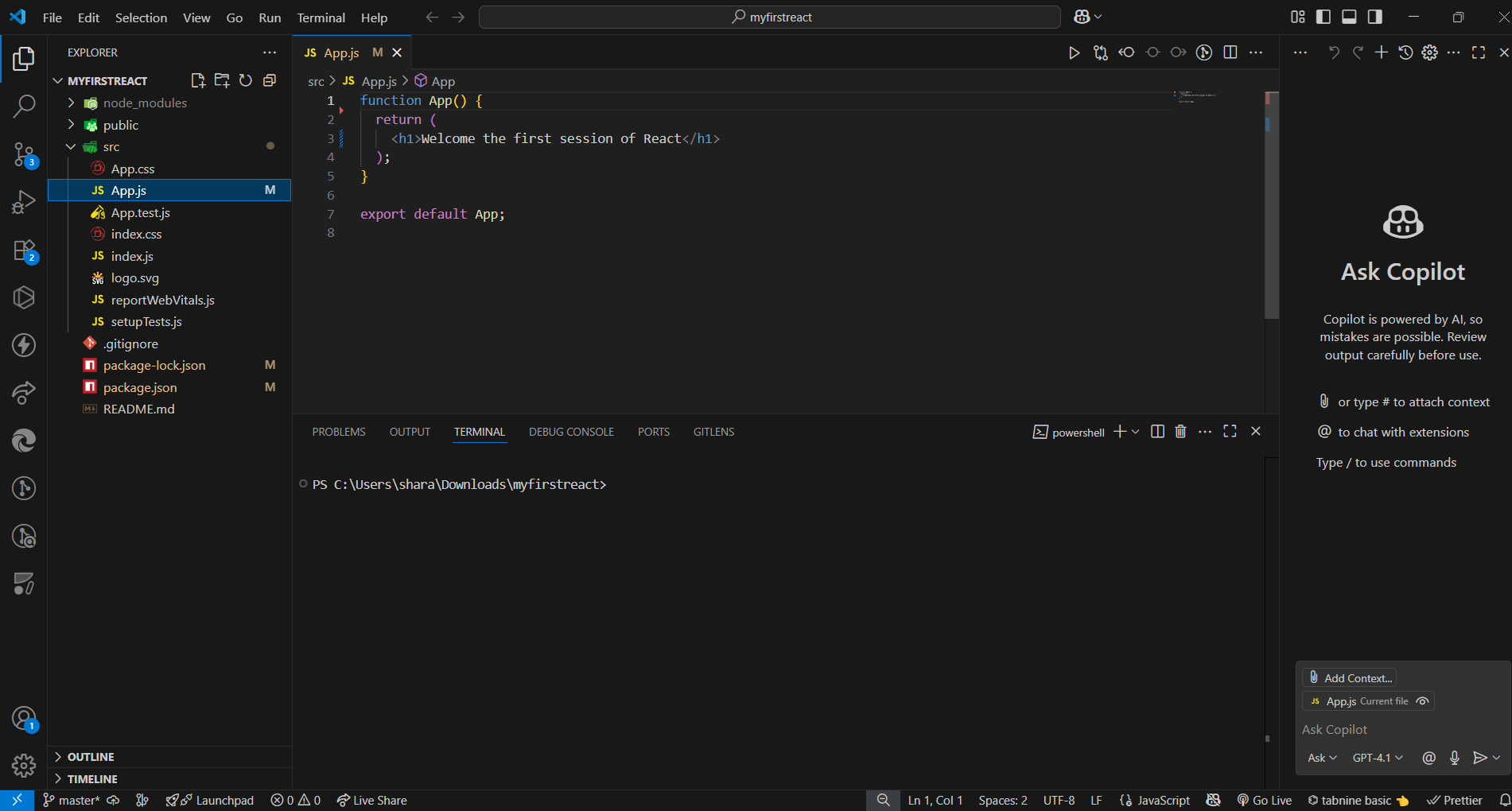
**ReactJS-HOL**

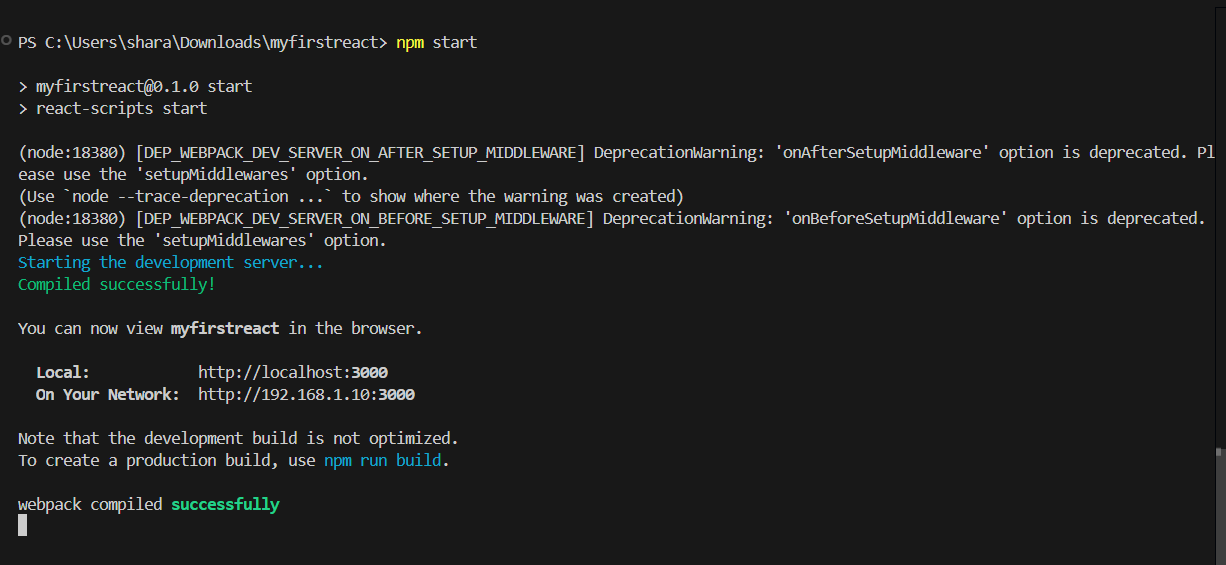
**Problem 1**

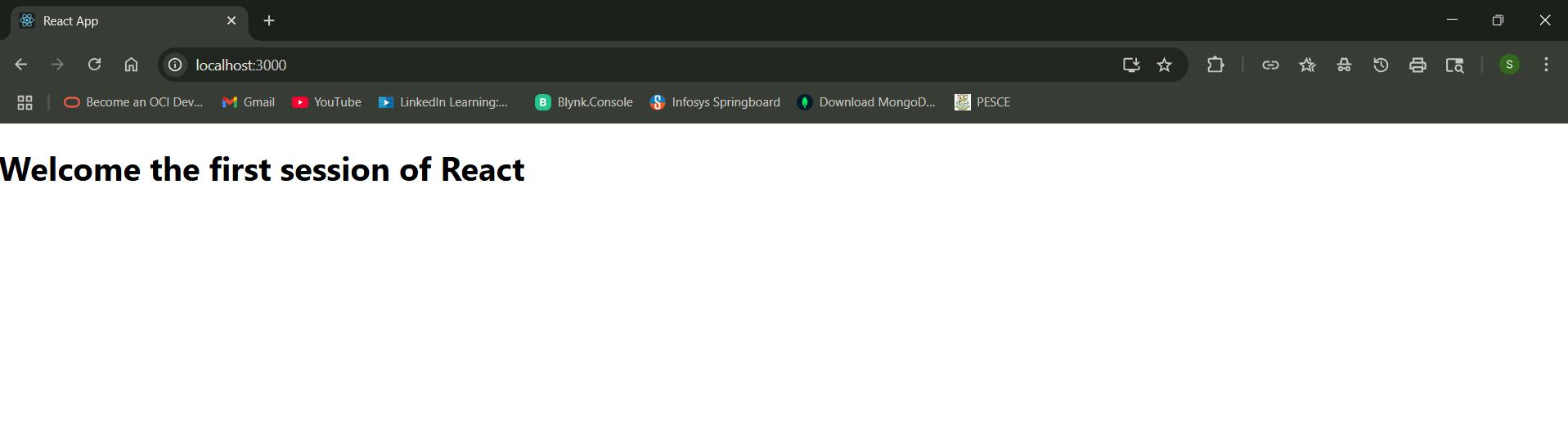
Create a new React Application with the name “myfirstreact”, Run the application to print “welcome to the first session of React” as heading of that page.

**Solution:**





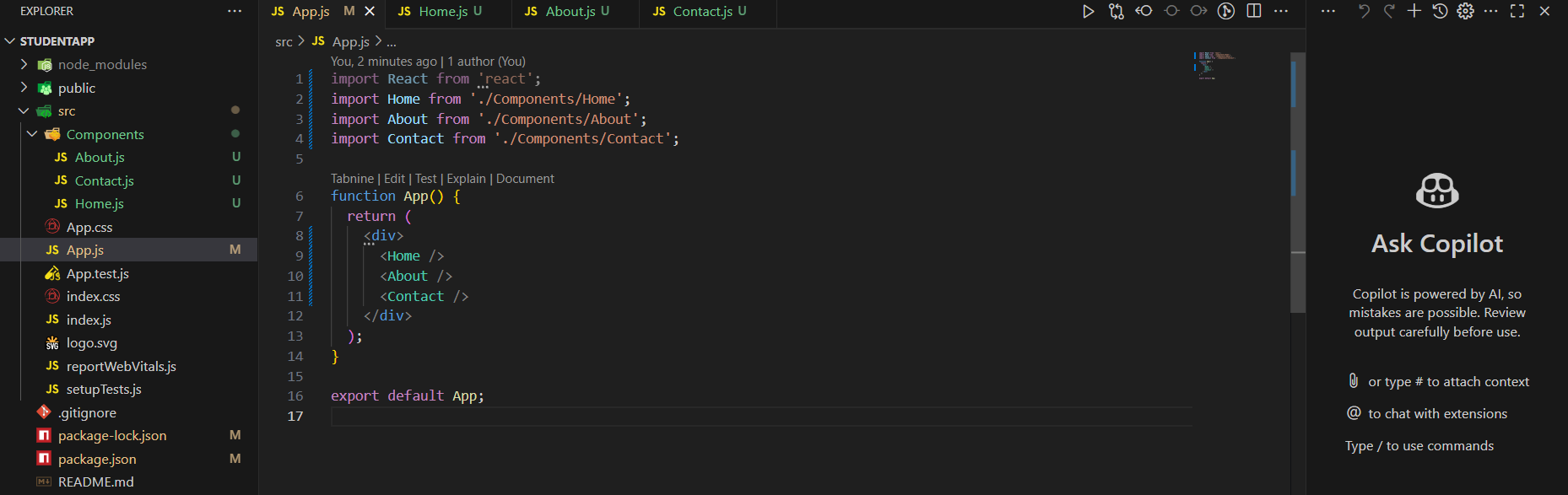




**Problem 2:**

Create a react app for Student Management Portal named StudentApp and create a component named Home which will display the Message “Welcome to the Home page of Student Management Portal”. Create another component named About and display the Message “Welcome to the About page of the Student Management Portal”. Create a third component named Contact and display the Message “Welcome to the Contact page of the Student Management Portal”. Call all the three components.

**Solution:**

****

**Home.js**

import React from 'react';

function Home() {

  return (

    <div>

      <h1>Welcome to the Home page of Student Management Portal</h1>

    </div>

  );

}

export default Home;

**About.js**

import React from 'react';

function About() {

  return (

    <div>

      <h1>Welcome to the About page of the Student Management Portal</h1>

    </div>

  );

}

export default About;

**Contact.js**

import React from 'react';

function Contact() {

  return (

    <div>

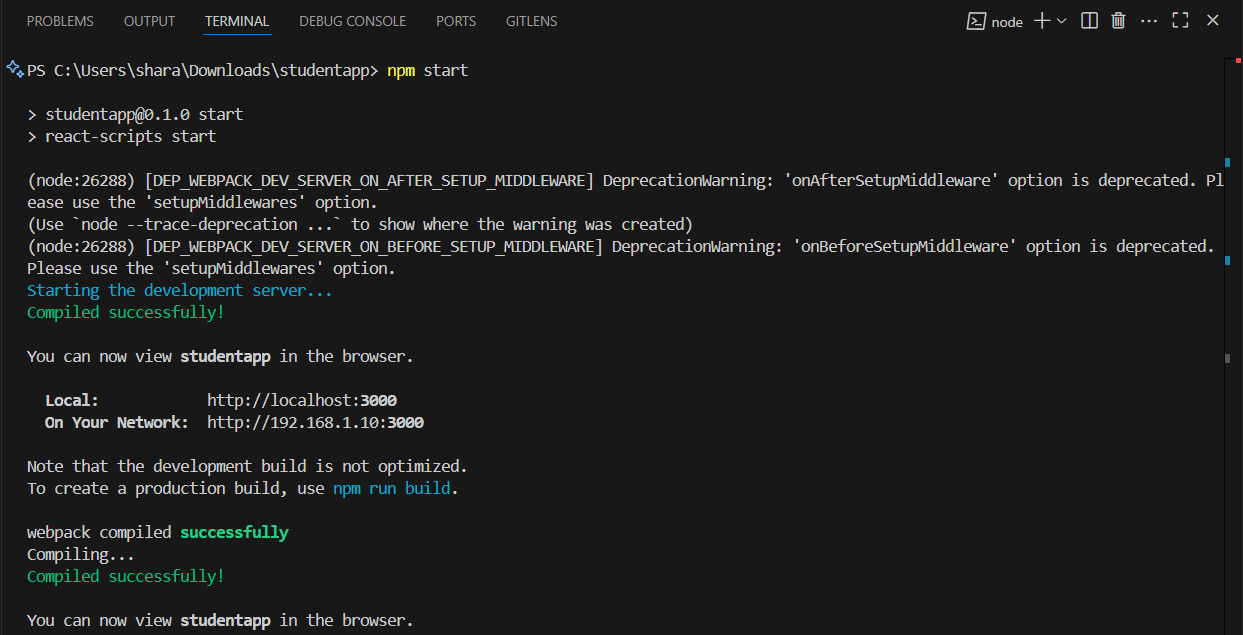
      <h1>Welcome to the Contact page of the Student Management Portal</h1>

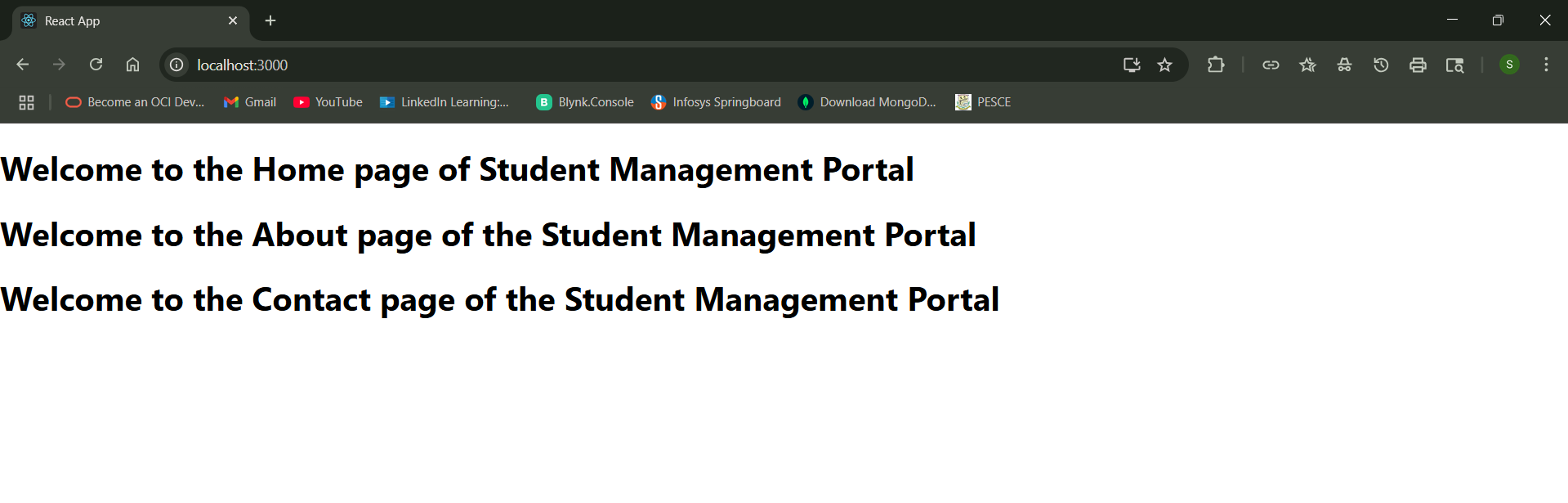
    </div>

  );

}

export default Contact;





**Problem 3:**

Create a react app for Student Management Portal named scorecalculatorapp and create a function component named “CalculateScore” which will accept Name, School, Total and goal in order to calculate the average score of a student and display the same.

**Solution:**

**Calculatescore.js**

import './Stylesheets/mystyle.css'

const percentToDecimal = (decimal) => {

return (decimal.toFixed(2) + "%")

}

const calcScore = (total, goal) => {

return percentToDecimal(total / goal)

}

export const CalculateScore = ({Name, School, total, goal}) => {

return (

<div className="formatstyle">

<h1><font color="Brown">Student Details:</font></h1>

<div className="Name">

<b><span> Name: </span></b>

<span>{Name}</span>

</div>

<div className="School">

<b><span> School: </span></b>

<span>{School}</span>

</div>

<div className="Total">

<b><span>Total: </span></b>

<span>{total}</span>

<span> Marks</span>

</div>

<div className="Score">

<b>Score:</b>

<span>

{calcScore(

total,

goal

)}

</span>

</div>

</div>

)

}

**Mystyle.css**

.Name {

font-weight: 300;

color: blue;

}

.School {

color: crimson;

}

.Total {

color: darkmagenta;

}

.formatstyle {

text-align: center;

font-size: large;

}

.Score {

color: forestgreen;

}

**App.js**

import {CalculateScore} from './Components/CalculateScore';

function App() {

    return (

        <div>

            <CalculateScore Name={"Sharanya K"}

                            School={"PES College of Engineering"}

                            total={284}

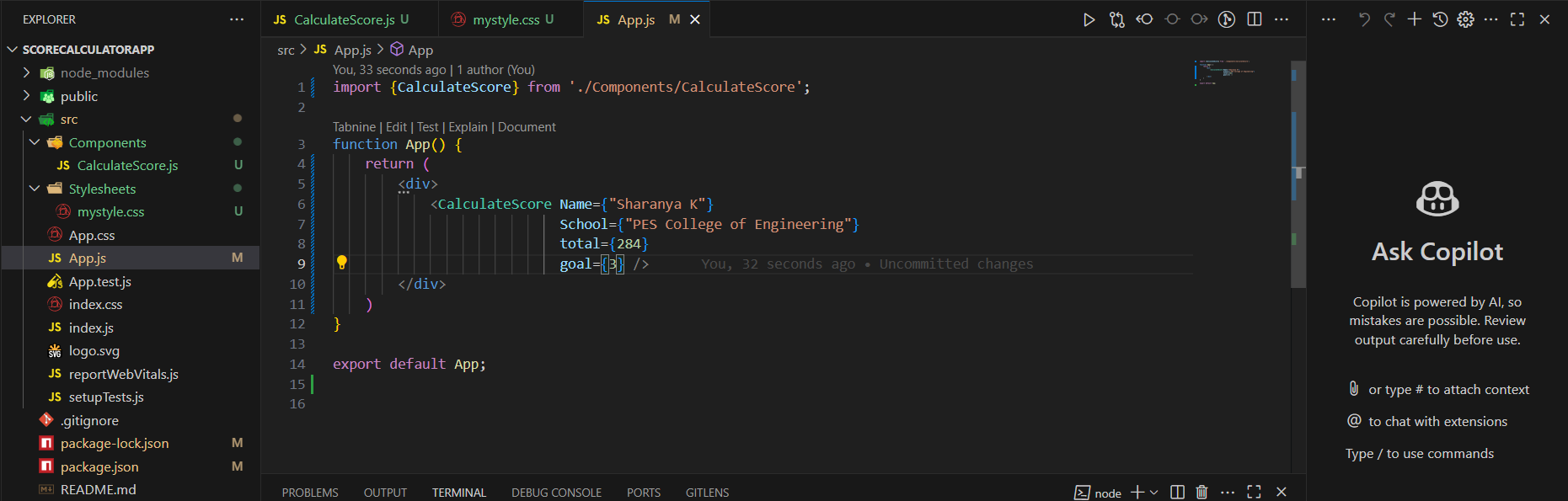
                            goal={3} />

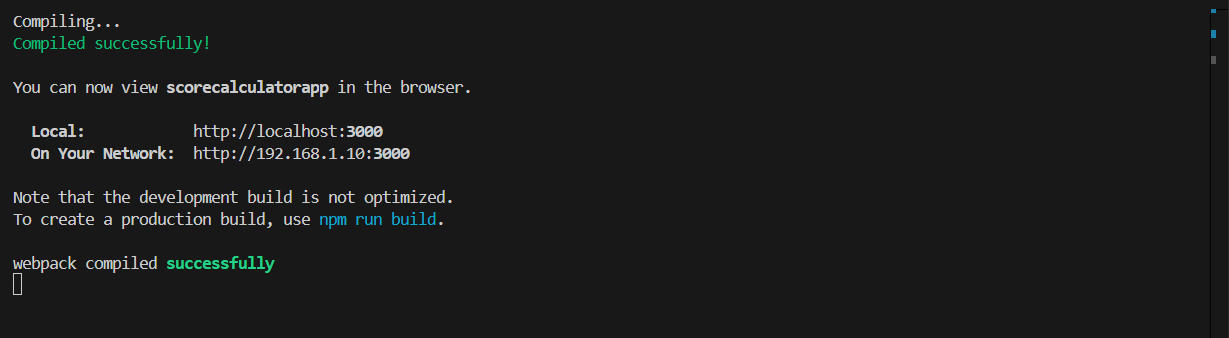
        </div>

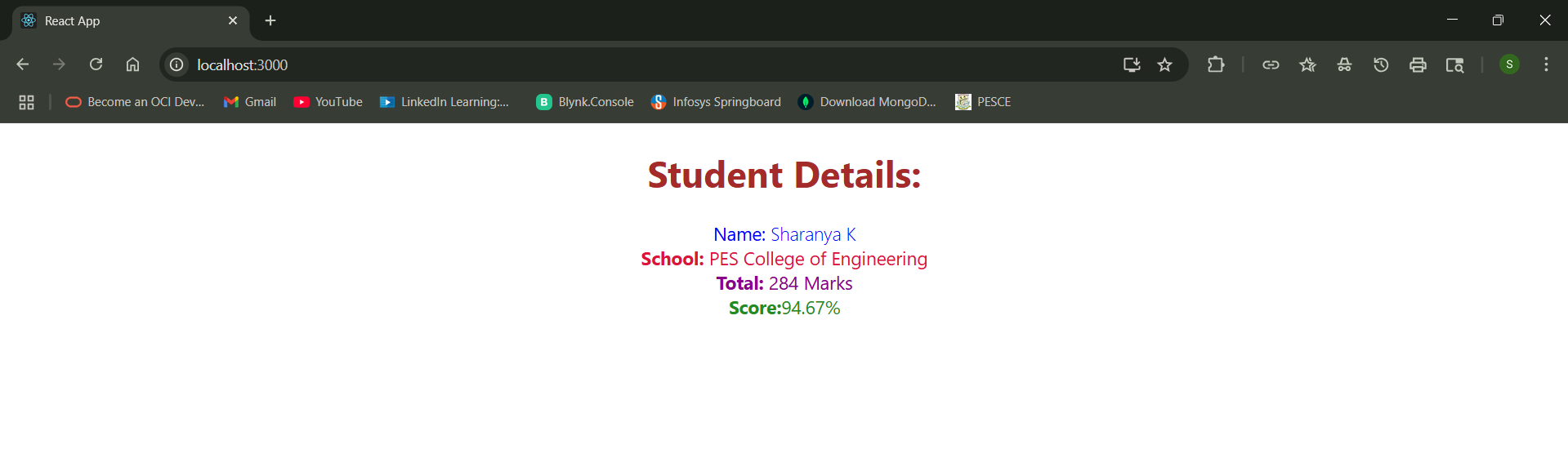
    )

}

export default App;







**Problem 4:**

1. Create a new react application using create-react-app tool with the name as “blogapp”

2. Open the application using VS Code

3. Create a new file named as Post.js in src folder with following properties

4. Create a new class based component named as Posts inside Posts.js file

5. Initialize the component with a list of Post in state of the component using the constructor

6. Create a new method in component with the name as loadPosts() which will be responsible for using Fetch API and assign it to the component state created earlier. To get the posts use the url (https://jsonplaceholder.typicode.com/posts)

7. Implement the componentDidMount() hook to make calls to loadPosts() which will fetch the posts

8. Implement the render() which will display the title and post of posts in html page using heading and paragraphs respectively.

9. Define a componentDidCatch() method which will be responsible for displaying any error happing in the component as alert messages.

10. Add the Posts component to App component.

11. Build and Run the application using npm start command.

**Solution:**

**Post.js**

class Post {

    constructor(id, title, body) {

        this.id = id;

        this.title = title;

        this.body = body;

    }

}

export default Post;

**Posts.js**

import React from 'react';

import Post from './Post';

class Posts extends React.Component {

    constructor(props) {

        super(props);

        this.state = {

            posts: [],

            error: null

        };

    }

    loadPosts() {

        fetch('https://jsonplaceholder.typicode.com/posts')

            .then(response => response.json())

            .then(data => {

                const postsList = data.map(item => new Post(item.id, item.title, item.body));

                this.setState({ posts: postsList });

            })

            .catch(error => this.setState({ error: error.message }));

    }

    componentDidMount() {

        this.loadPosts();

    }

    componentDidCatch(error, info) {

        alert('Error occurred: ' + error);

    }

    render() {

        return (

            <div>

                <h1>Posts</h1>

                {this.state.posts.length === 0 ? (

                    <p>Loading posts...</p>

                ) : (

                    this.state.posts.map(post => (

                        <div key={post.id}>

                            <h3>{post.title}</h3>

                            <p>{post.body}</p>

                        </div>

                    ))

                )}

            </div>

        );

    }

}

export default Posts;

**App.js**

import React from 'react';

import Posts from './Posts';

function App() {

    return (

        <div>

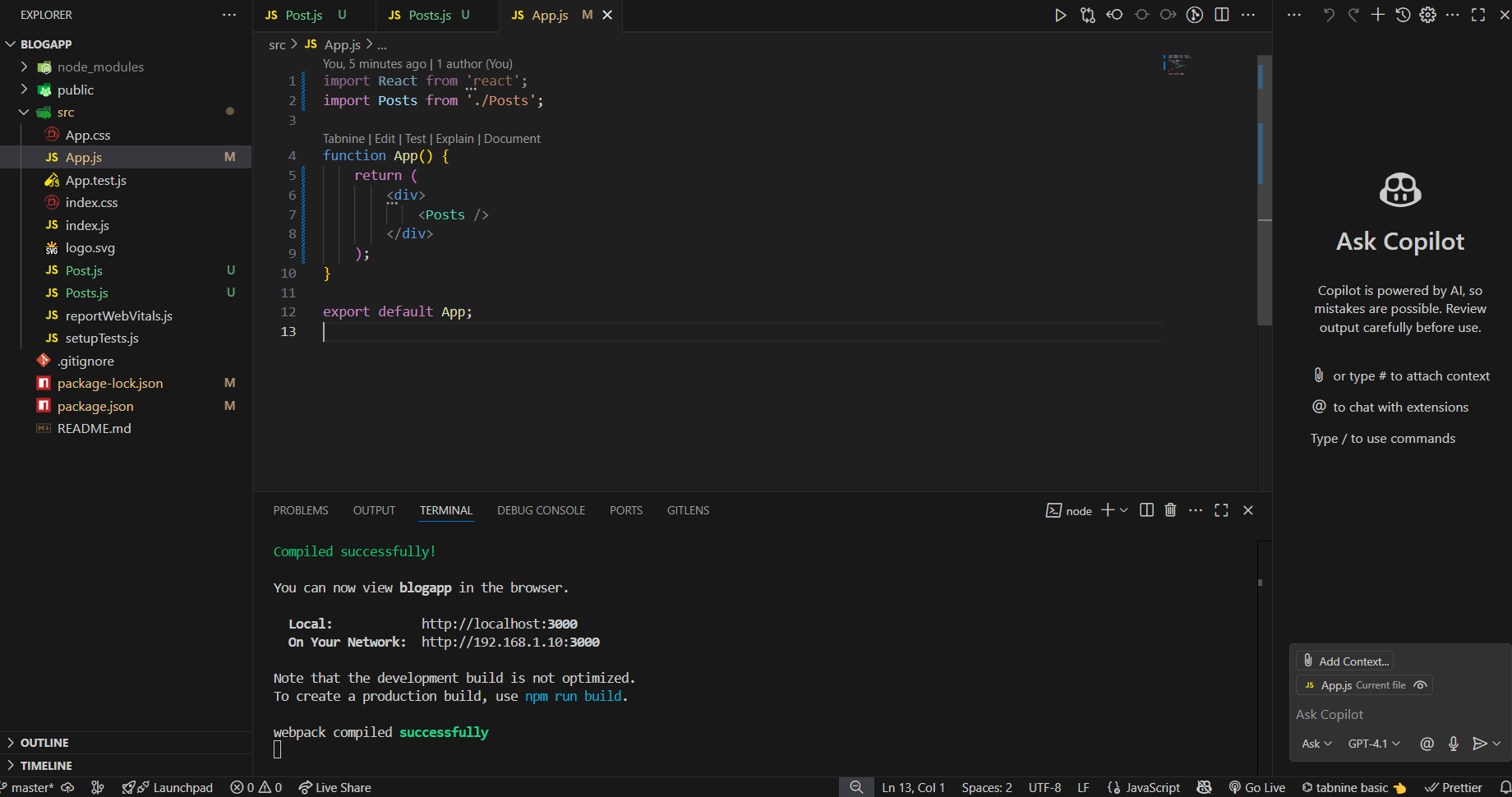
            <Posts />

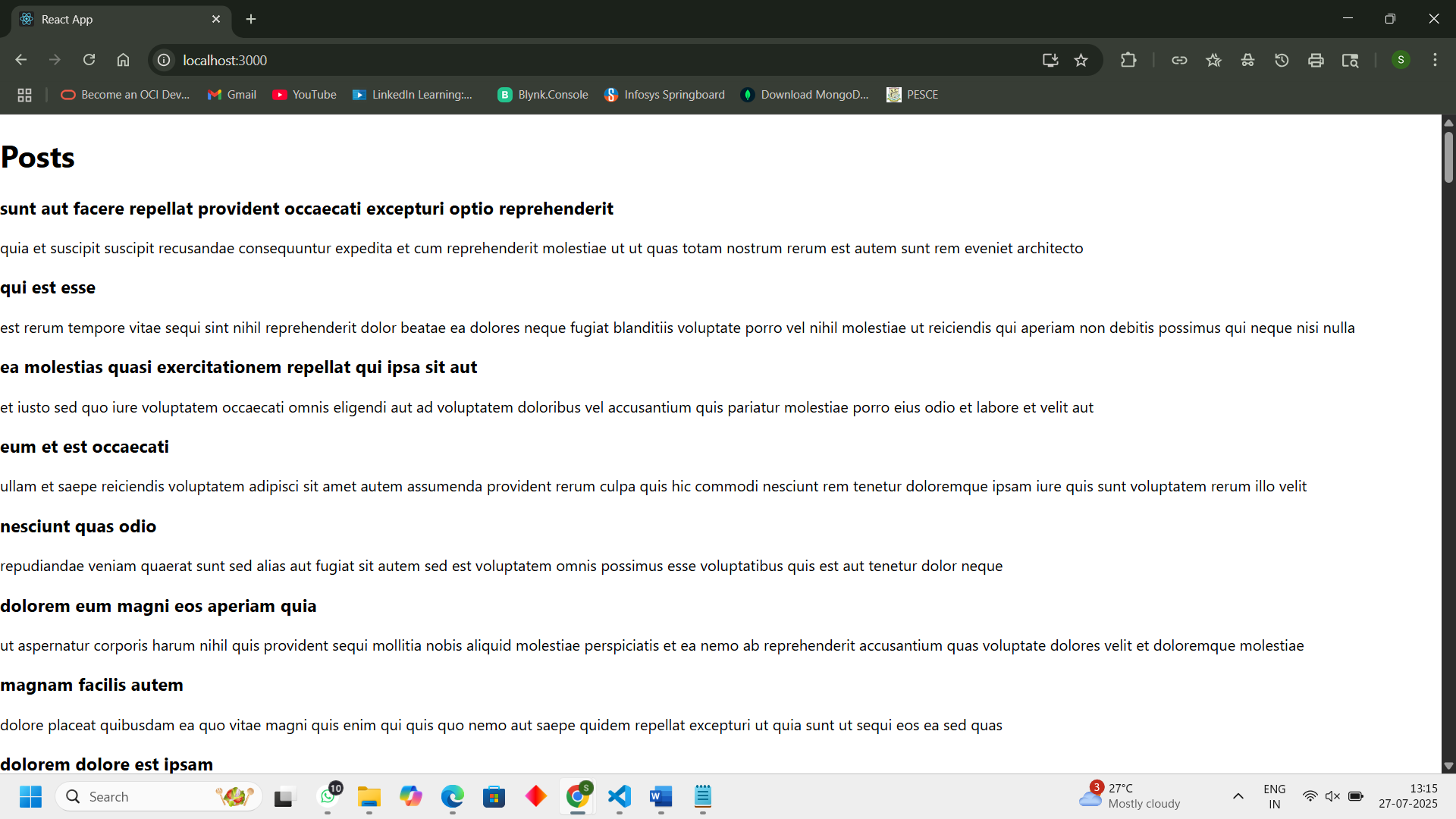
        </div>

    );

}

export default App;





**Problem 5:**

My Academy team at Cognizant want to create a dashboard containing the details of ongoing and completed cohorts. A react application is created which displays the detail of the cohorts using react component. You are assigned the task of styling these react components.

**Solution:**

**CohortDetails.module.css**

.box {

    width: 300px;

    display: inline-block;

    margin: 10px;

    padding: 10px 20px;

    border: 1px solid black;

    border-radius: 10px;

}

dt {

    font-weight: 500;

}

**CohortDetails.js**

import React from 'react';

import styles from './CohortDetails.module.css';

function CohortDetails(props) {

// Conditional inline style for <h3>

const headingStyle = {

color: props.cohort.currentStatus.toLowerCase() === 'ongoing' ? 'green' : 'blue'

};

return (

<div className={styles.box}>

<h3 style={headingStyle}>

{props.cohort.cohortCode} -

<span>{props.cohort.technology}</span>

</h3>

<dl>

<dt>Started On</dt>

<dd>{props.cohort.startDate}</dd>

<dt>Current Status</dt>

<dd>{props.cohort.currentStatus}</dd>

<dt>Coach</dt>

<dd>{props.cohort.coachName}</dd>

<dt>Trainer</dt>

<dd>{props.cohort.trainerName}</dd>

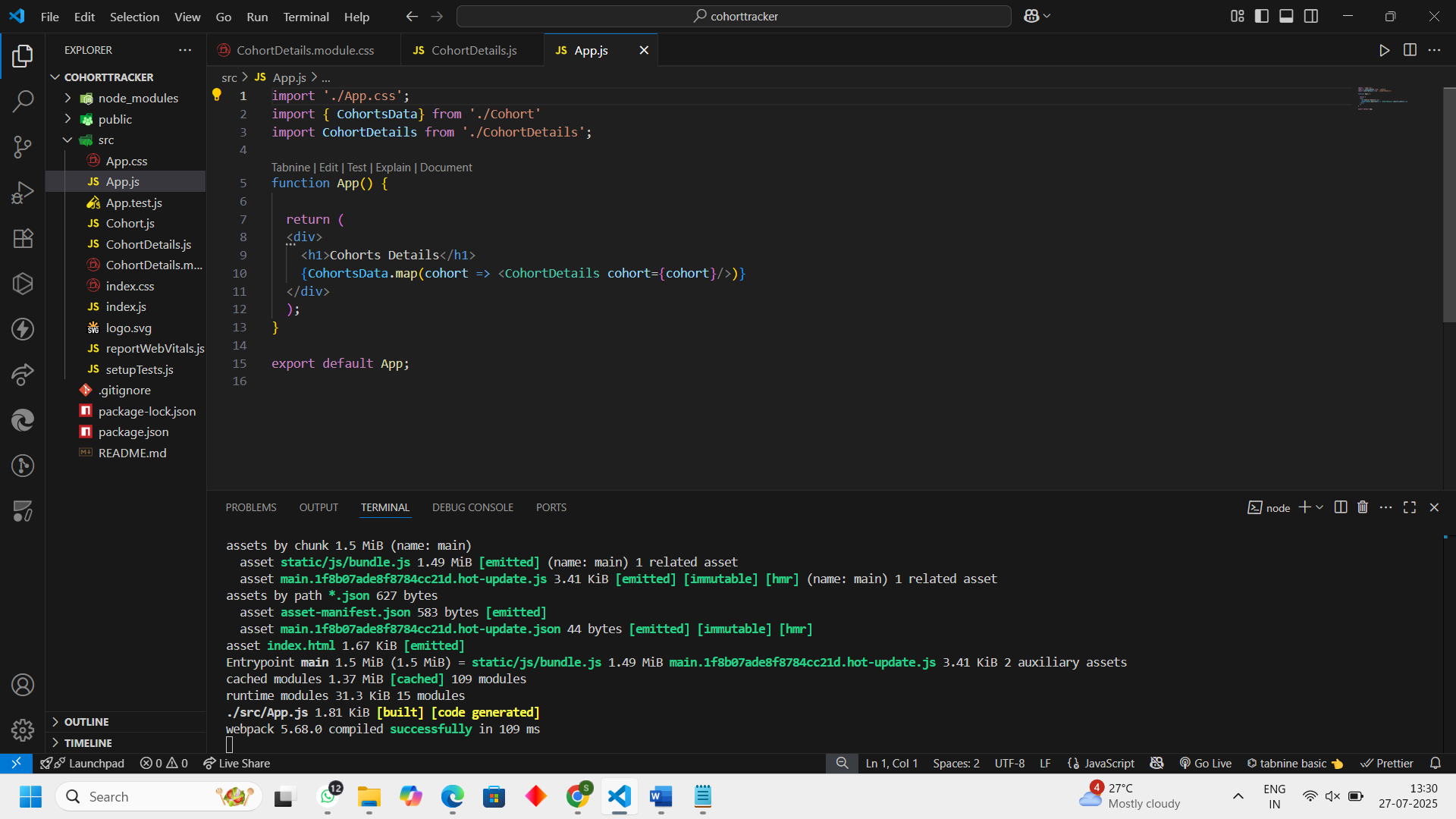
</dl>

</div>

);

}

export default CohortDetails;

****

