**(6428317 Mukesh P)**

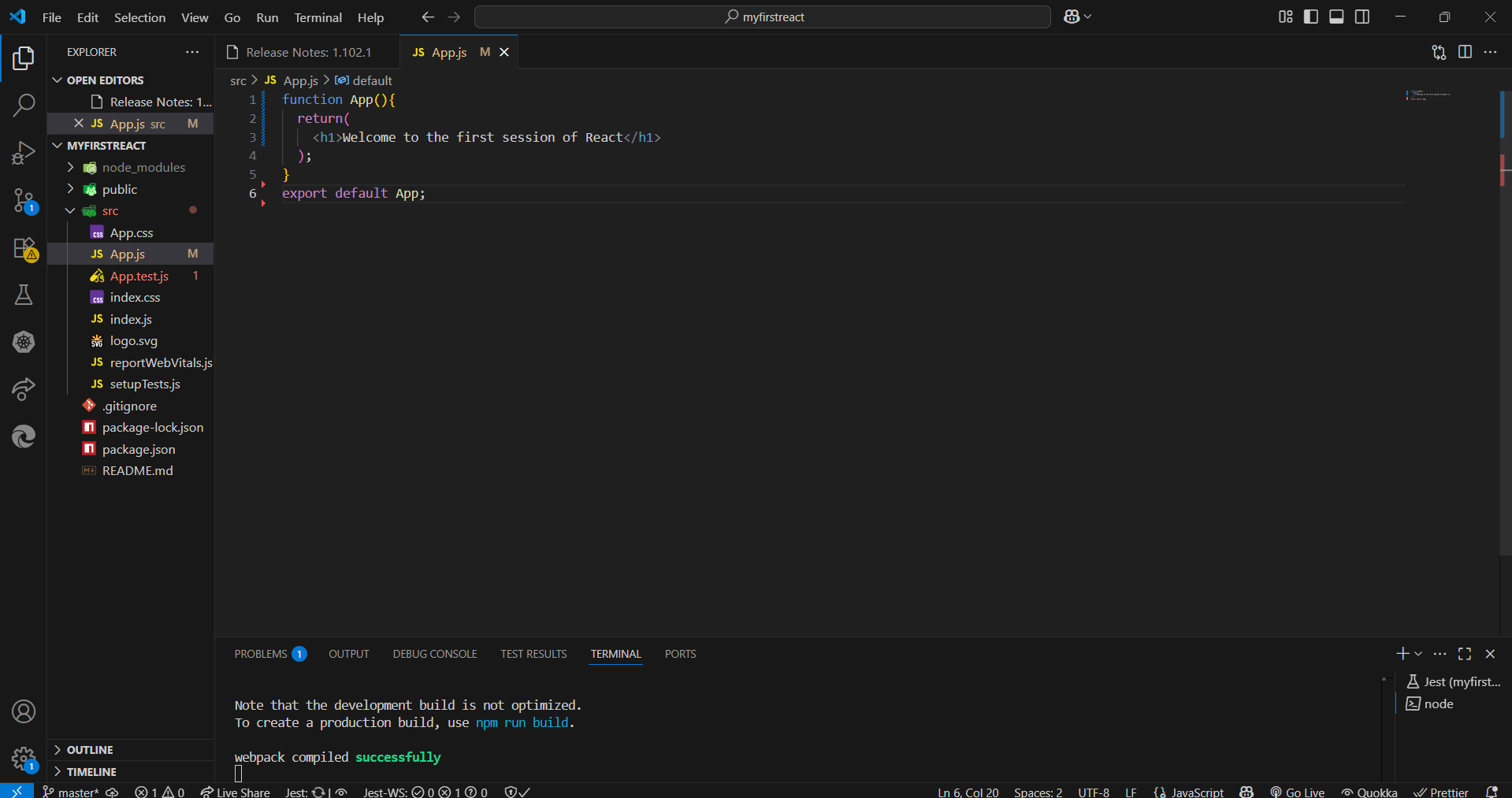
**WEEK 6 REACT**

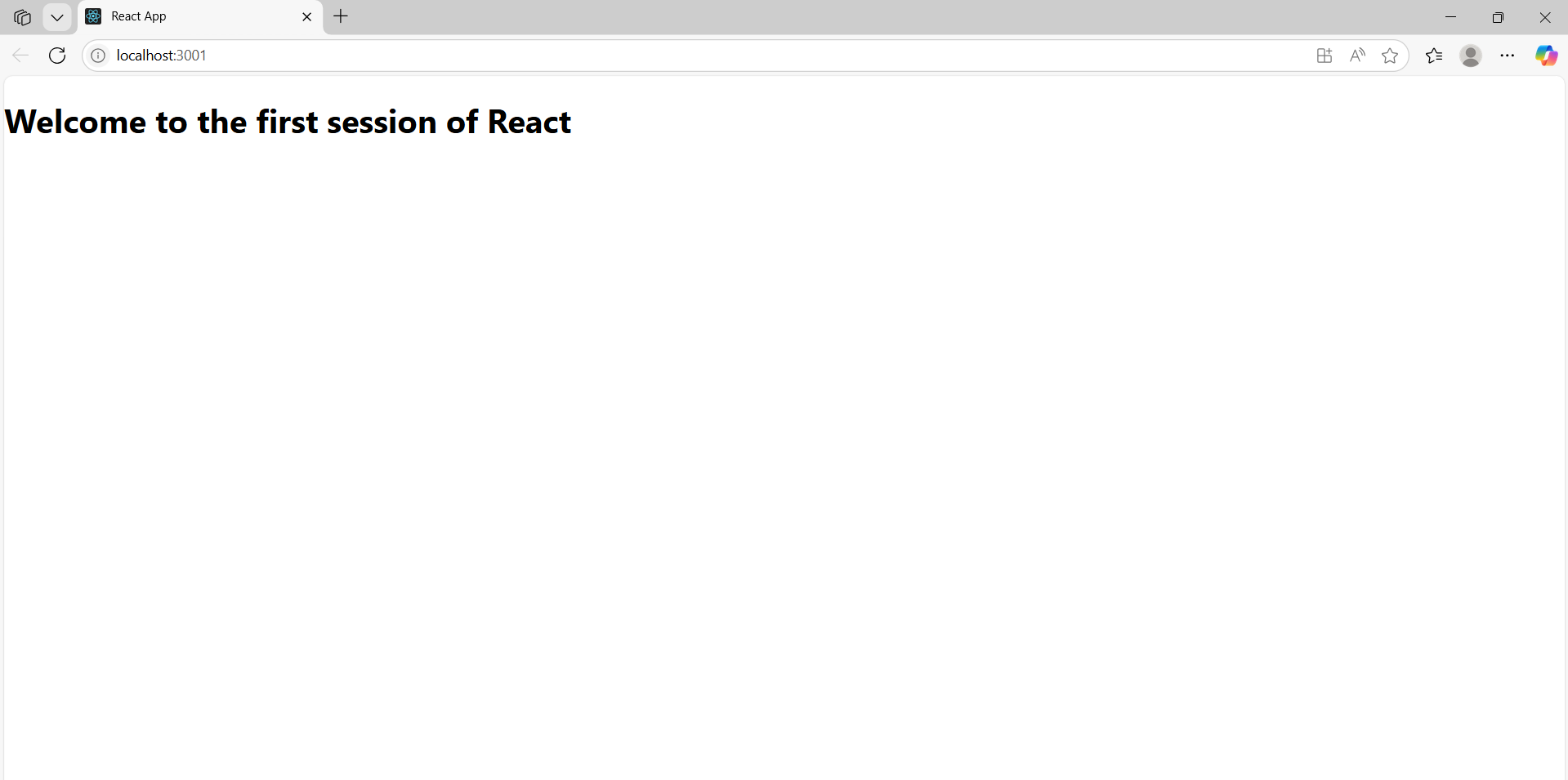
**Mandatory HandsOn:**

## **1.Objectives**

* Define SPA and its benefits
* Define React and identify its working
* Identify the differences between SPA and MPA
* Explain Pros & Cons of Single-Page Application
* Explain about React
* Define virtual DOM
* Explain Features of React

**O/P**

****

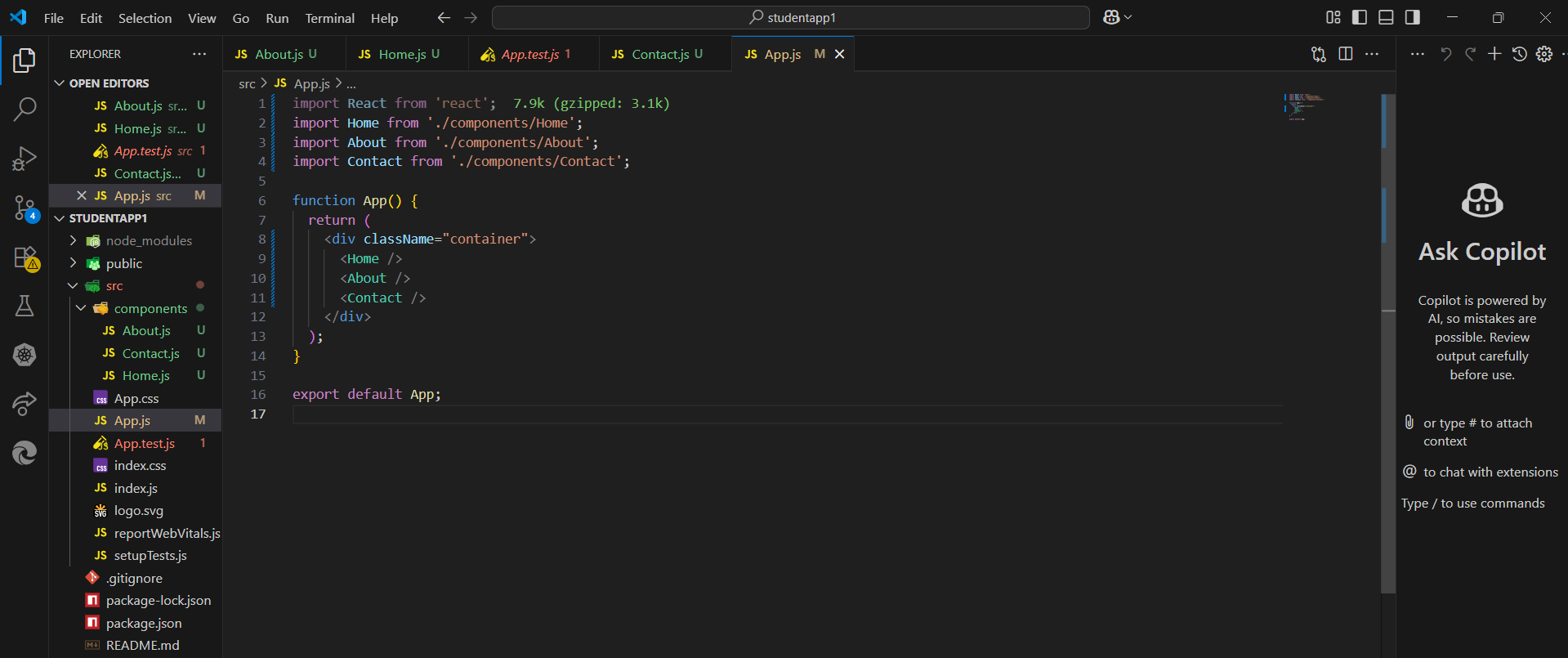


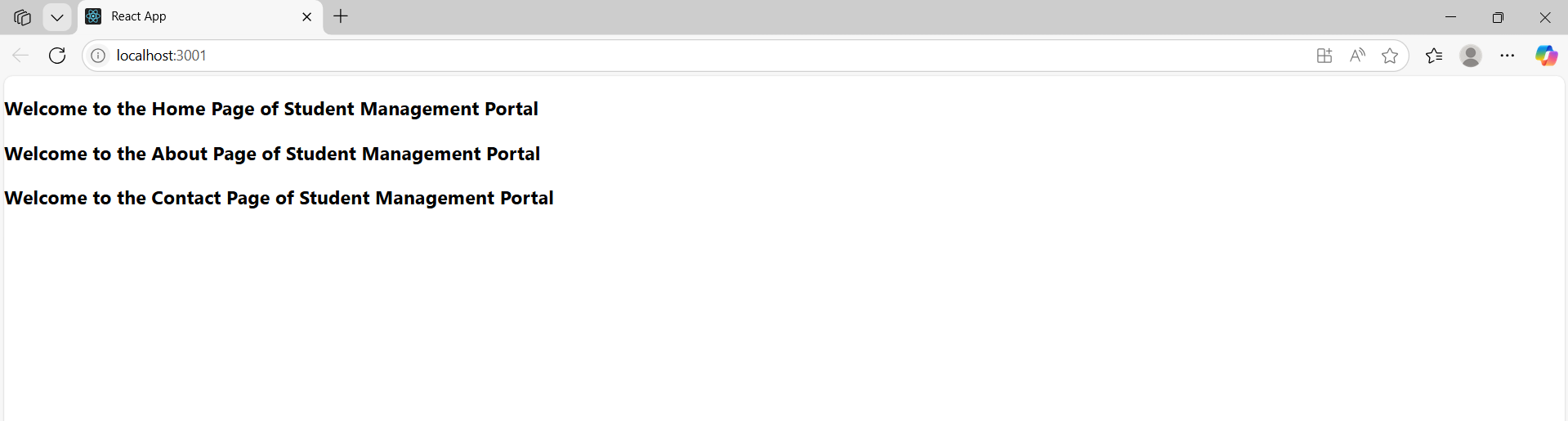
**2.REACT JS- HOL:**

## **Objectives**

* Explain React components
* Identify the differences between components and JavaScript functions
* Identify the types of components
* Explain class component
* Explain function component
* Define component constructor
* Define render() function

**O/P:**



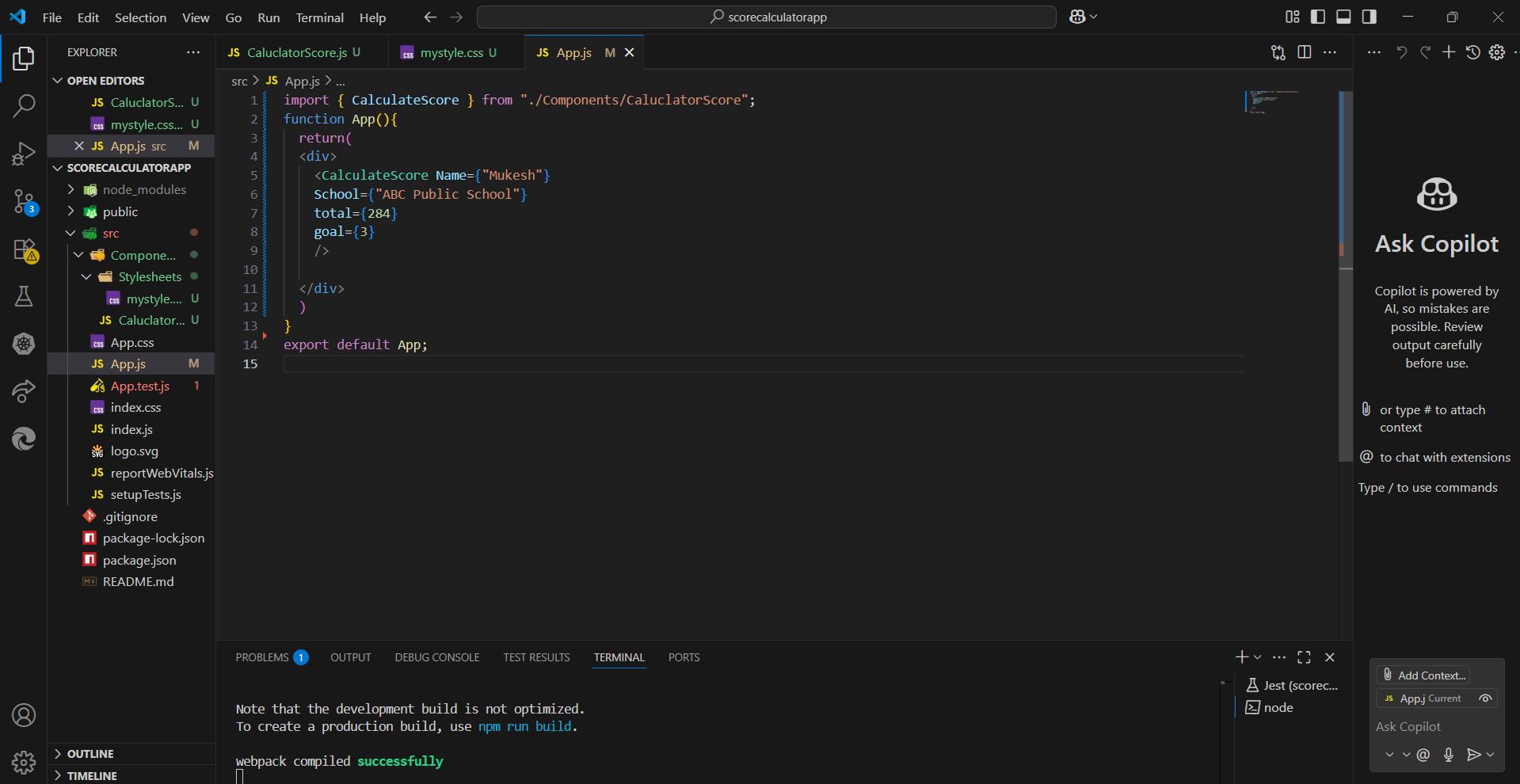


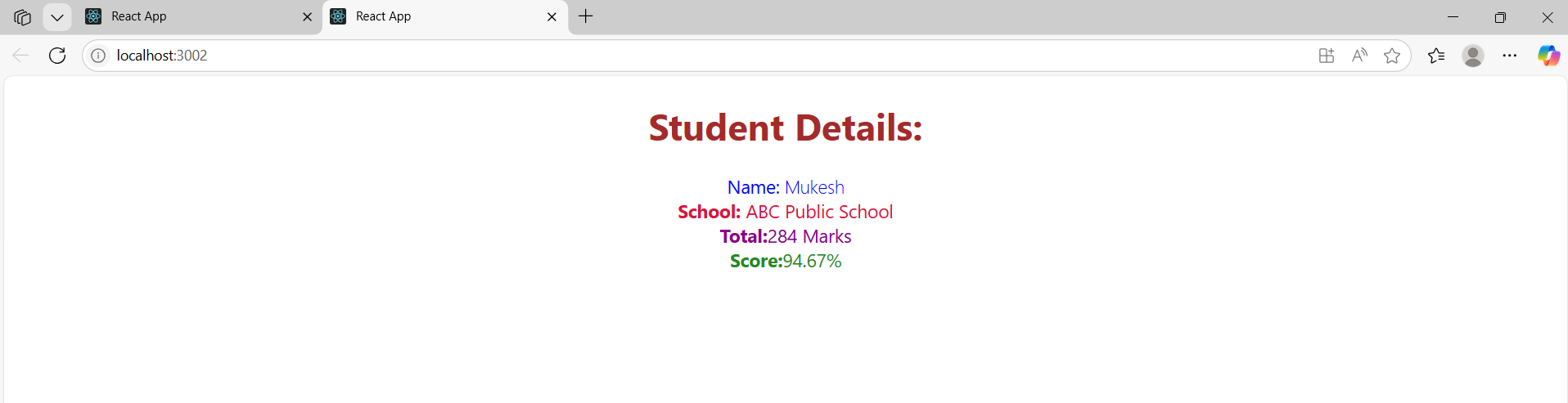
**3.REACT JS- HOL:**

**Objectives**

* **Explain React components**
* **Identify the differences between components and JavaScript functions**
* **Identify the types of components**
* **Explain class component**
* **Explain function component**
* **Define component constructor**
* **Define render() function**

**O/P:**





**4.REACT JS- HOL:**

**Objectives**

* Explain the need and Benefits of component life cycle
* Identify various life cycle hook methods
* List the sequence of steps in rendering a component

In this hands-on lab, you will learn how to:

* Implement componentDidMount() hook
* Implementing componentDidCatch() life cycle hook.

import React from 'react';

const Post = ({ title, body }) => {

  return (

    <div style={{ marginBottom: '20px' }}>

      <h2>{title}</h2>

      <p>{body}</p>

    </div>

  );

};

export default Post;

import React, { Component } from 'react';

import Post from './Post';

class Posts extends Component {

  constructor(props) {

    super(props);

    this.state = {

      posts: [],

      hasError: false

    };

  }

  loadPosts = () => {

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

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

      .then(data => this.setState({ posts: data }))

      .catch(error => {

        console.error('Error fetching posts:', error);

        this.setState({ hasError: true });

      });

  };

  componentDidMount() {

    this.loadPosts();

  }

  componentDidCatch(error, info) {

    alert('Something went wrong!');

    console.error("Error boundary caught an error:", error, info);

  }

  render() {

    return (

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

        <h1>Blog Posts</h1>

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

          <Post key={post.id} title={post.title} body={post.body} />

        ))}

      </div>

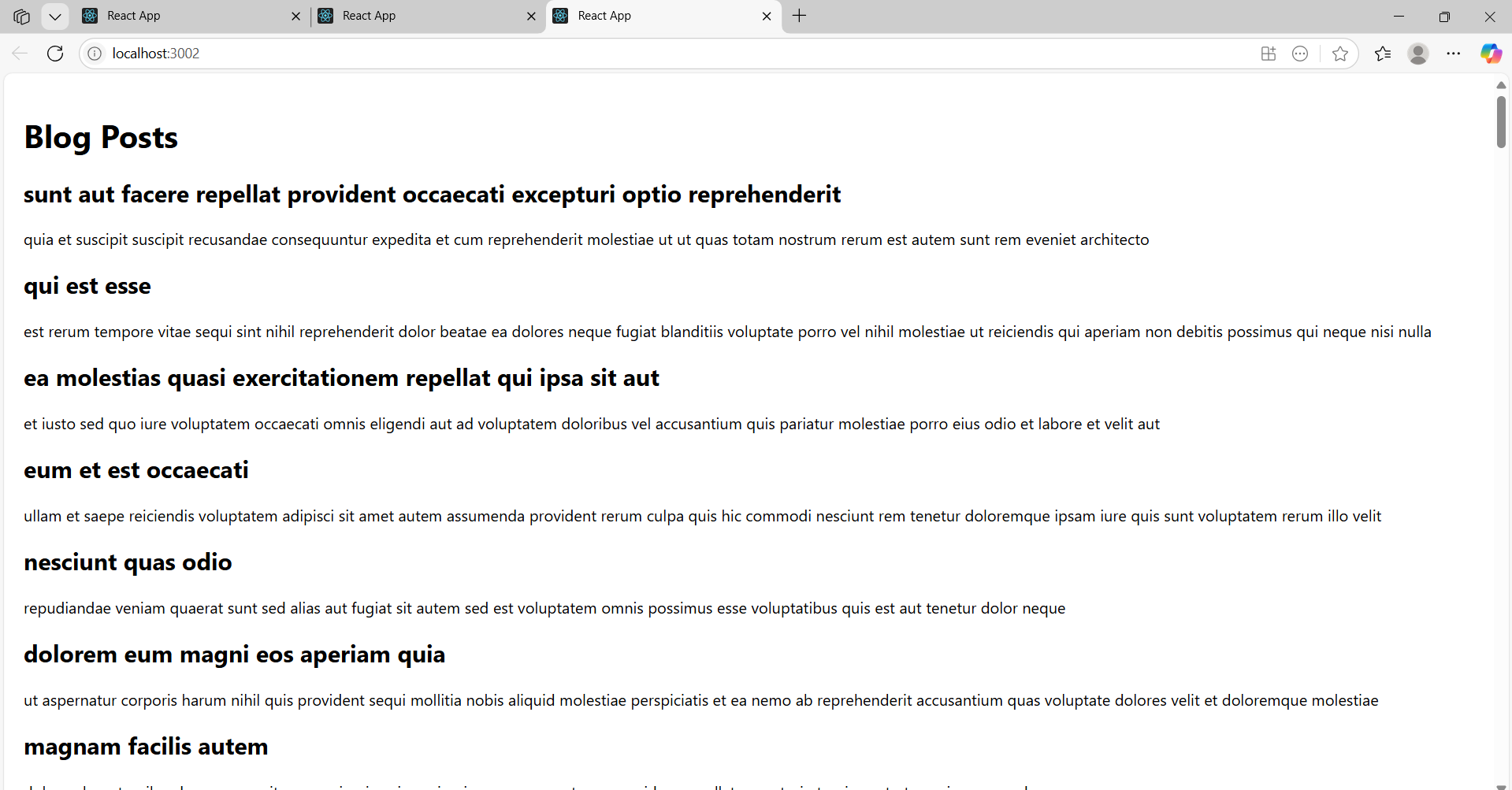
    );

  }

}

export default Posts;

**O/P:**



**5.REACT JS- HOL:**

**Objectives**

* Understanding the need for styling react component
* Working with CSS Module and inline styles

In this hands-on lab, you will learn how to:

* Style a react component
* Define styles using the CSS Module
* Apply styles to components using className and style properties

.box {

  width: 300px;

  display: inline-block;

  margin: 10px;

  padding: 10px 20px;

  border: 1px solid black;

  border-radius: 10px;

}

dt {

  font-weight: 500;

}

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

function CohortDetails(props) {

  const statusColor = props.cohort.currentStatus === 'Ongoing' ? 'green' : 'blue';

  return (

    <div className={styles.box}>

      <h3 style={{ color: statusColor }}>

        {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;

import './App.css';

import { CohortsData } from './Cohort';

import CohortDetails from './CohortDetails';

function App() {

  return (

    <div>

      <h1>Cohorts Details</h1>

      {CohortsData.map((cohort) => (

        <CohortDetails key={cohort.cohortCode} cohort={cohort} />

      ))}

    </div>

  );

}

export default App;

**O/P:**

