WEEK 6 MANDATORY HANDS-ON(REACT)

1. SPA (Single Page Application) is a web application that loads a single HTML page and dynamically updates content without reloading the entire page. It includes benefits such as faster navigation, reduced load times, a better user experience, and seamless interaction.
2. React is a JavaScript library for building user interfaces, mainly for single-page applications. **Working:** It uses a component-based architecture and updates the virtual DOM to efficiently render UI changes.

## Feature SPA MPA

Page Load Single page loaded once Multiple pages reloaded Speed Faster after initial load Slower due to page reloads UX Smooth transitions Flickers during navigation SEO Harder to optimize Easier SEO support

1. Advantages and disadvantages of SPA:

## Pros:

•

•

•

## Cons:

•

•

•

Fast and smooth UX Less bandwidth usage Reusable frontend code

Poor SEO by default Longer initial load time

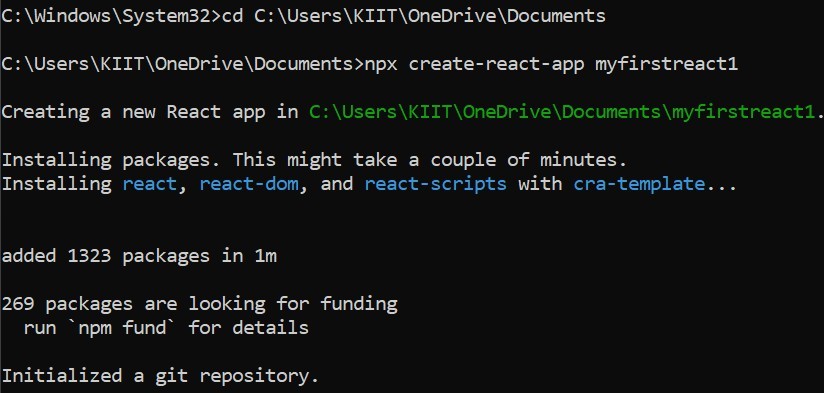
Complex state management

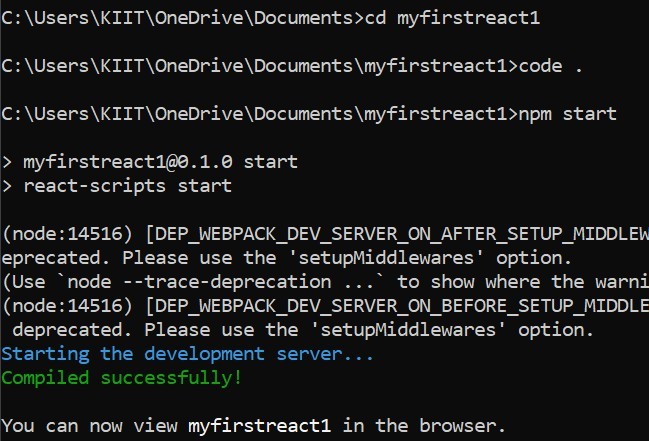
1. React is an open-source JavaScript library developed by Facebook to build dynamic and responsive UIs using components. It efficiently updates and renders components using a virtual DOM.
2. Virtual DOM is a lightweight in-memory representation of the real DOM. React updates the virtual DOM first, compares it with the previous version (diffing), and updates only the changed parts in the real DOM.
3. Features of React:
   * **Component-Based**: UI is built using reusable components
   * **JSX**: JavaScript + HTML syntax
   * **Virtual DOM**: Faster UI rendering

## Unidirectional Data Flow

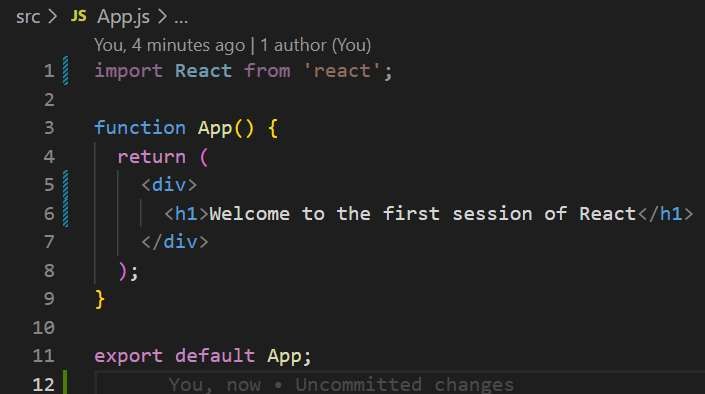
* + **Hooks**: Functional component state and lifecycle
  + **React Router**: For SPA navigation

Creating a new React Application with the name “myfirstreact1”. Running the application to print “Welcome to the first session of React” as the heading of that page.





App.js



OUTPUT:



# MODULE 2:

1. Explaining React components

React components are reusable building blocks used to define UI in React. They return JSX that describes how the UI should appear.

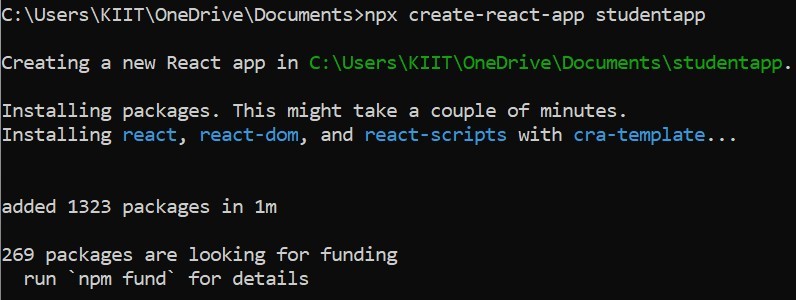
## React Component JavaScript Function

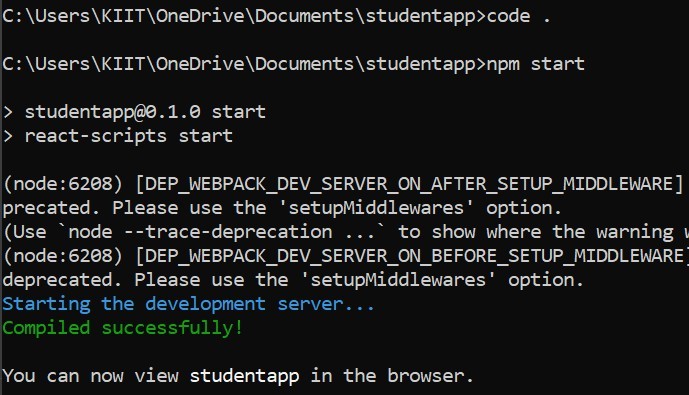
Returns JSX (UI) May return any value or nothing Used to render UI Used for general logic Lifecycle methods (class) No lifecycle methods React-specific behavior Purely logic-based

1. Types of components:

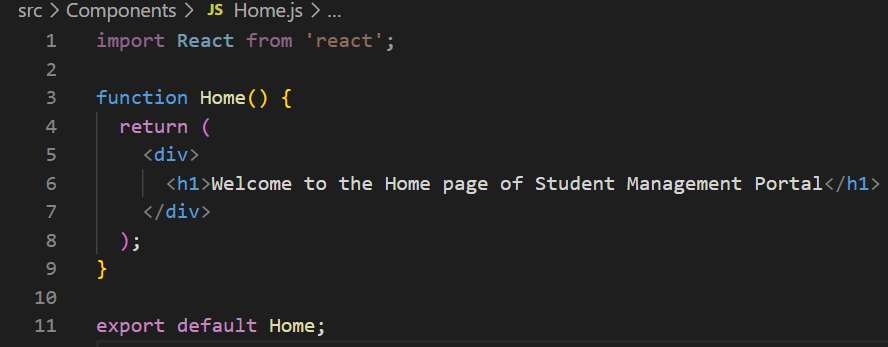
Class component Function component

1. A class component is a React component defined using an ES6 class. It can use state, lifecycle methods, and must include a render() method.
2. A function component is a plain JavaScript function that returns JSX. It can use React Hooks to handle state and lifecycle logic.
3. In a class component, the constructor() is used to initialize state and bind methods. It runs before the component is mounted.
4. The render() function is **required in** class components. It returns JSX that defines what should be displayed in the UI.

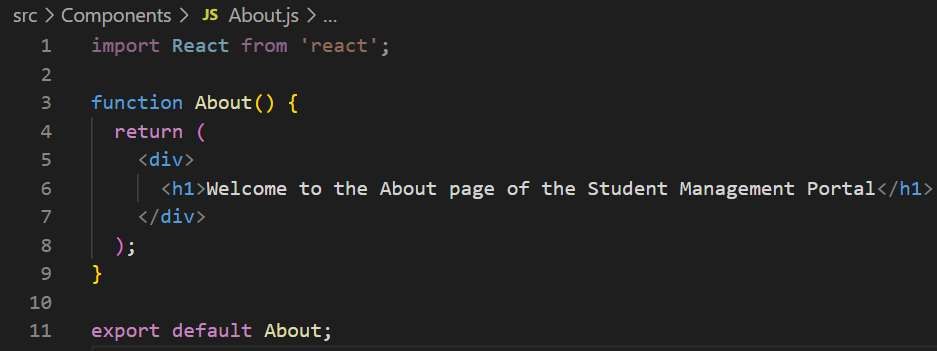




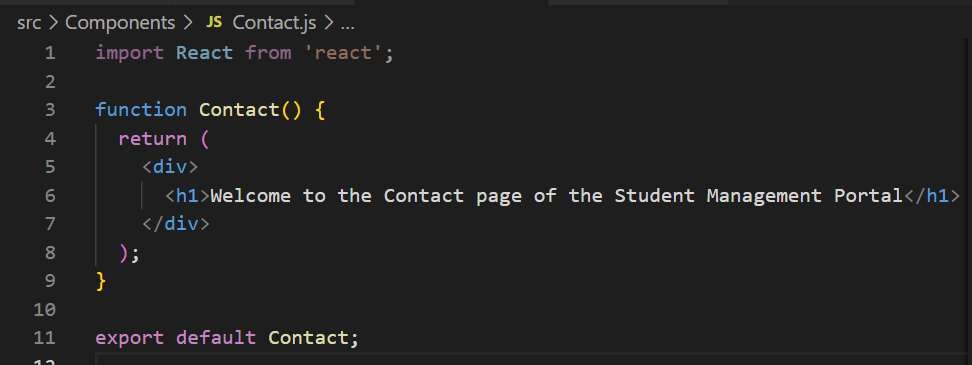
Home.js



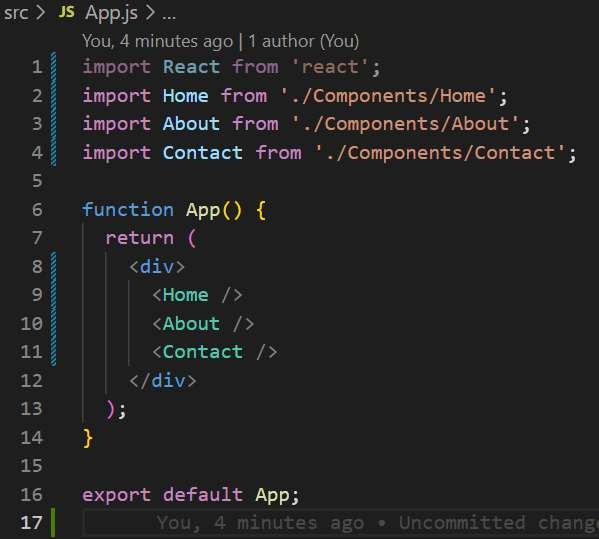
About.js



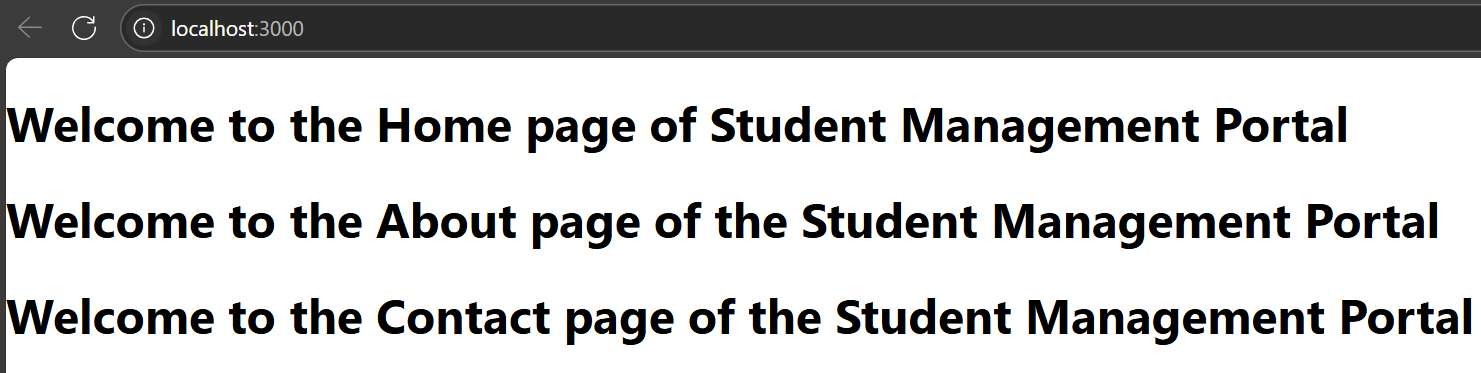
Contact.js



App.js

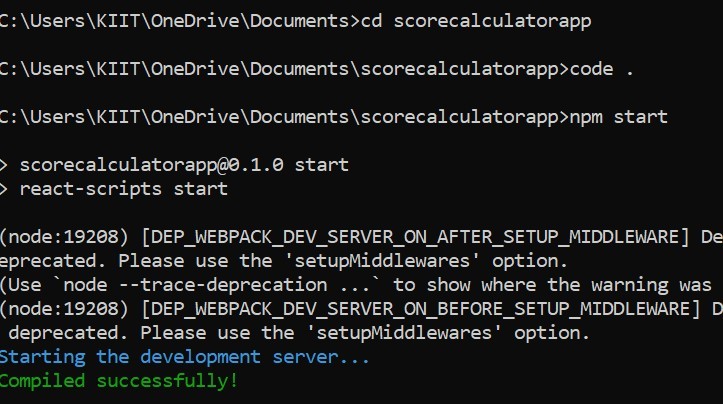
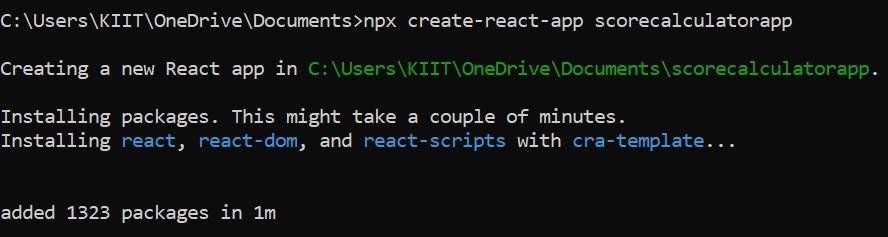


OUTPUT:

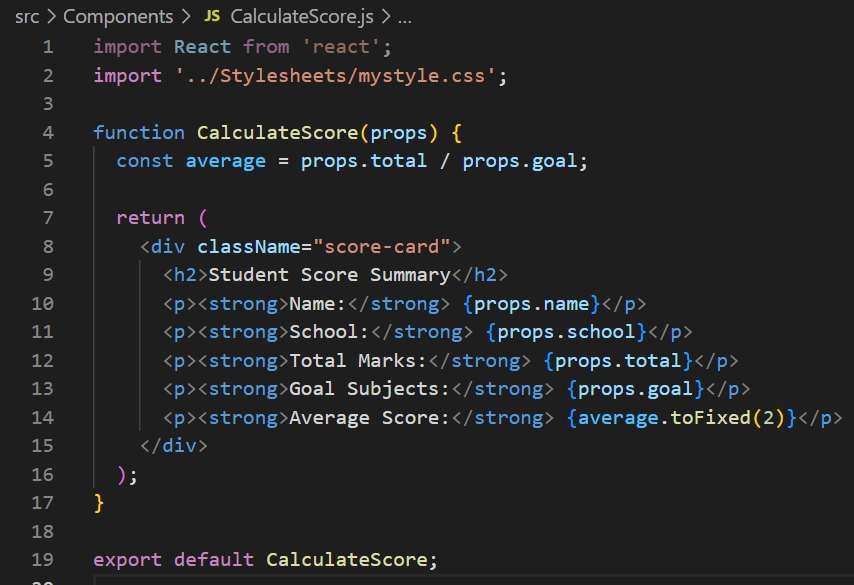


# MODULE 3:

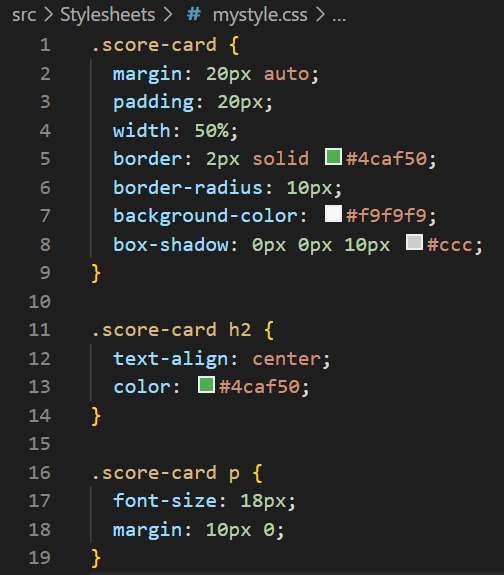
Create a React app for Student Management Portal named scorecalculatorapp and create a function component named “CalculateScore” which will accept name, school, total, and goal to calculate the average score of a student and display the same.



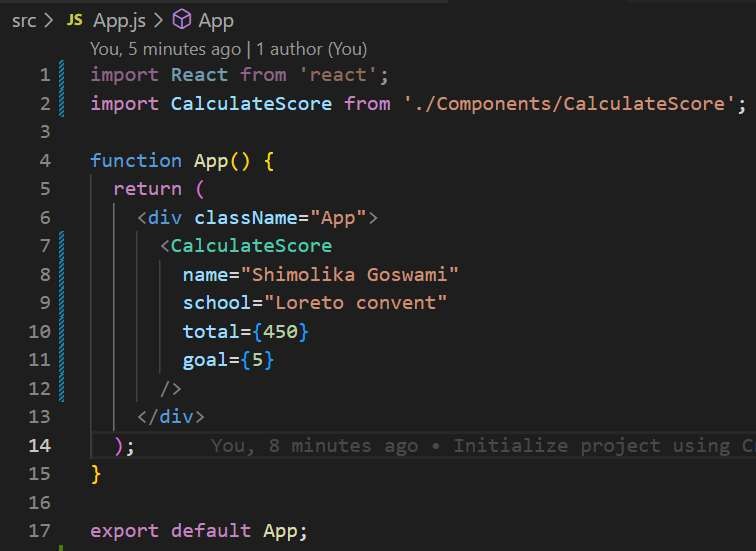
CalculateScore.js



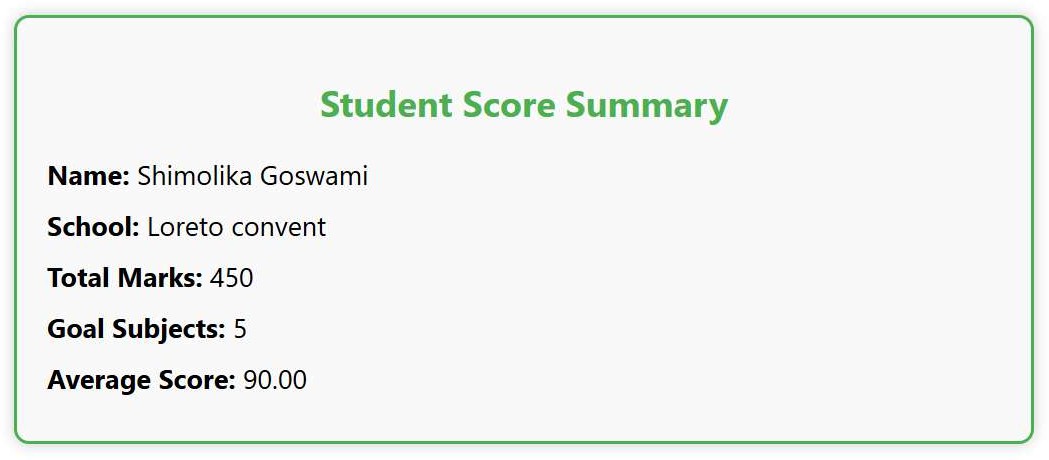
Mystyle.css



App.js



OUTPUT:



# MODULE 4:

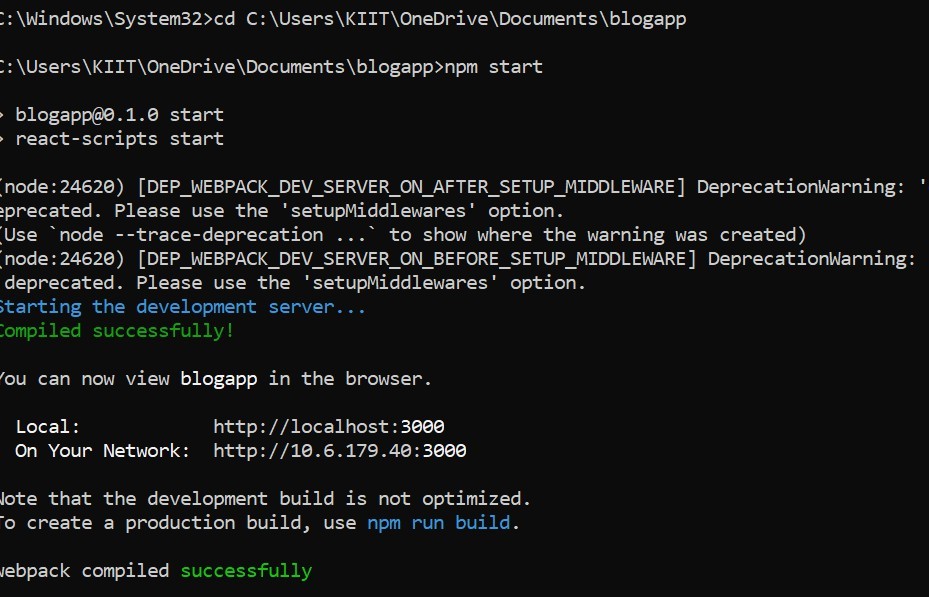
Component lifecycle allows you to control what happens **before**, **during**, and **after** a component is rendered in the DOM.

Benefits:

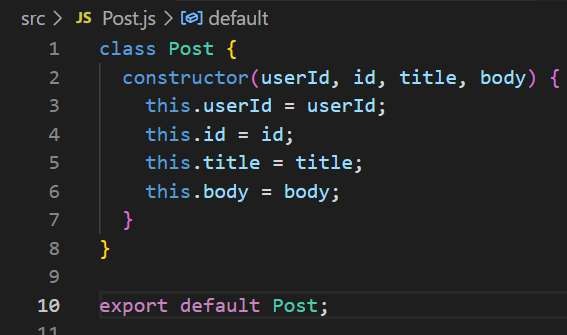
* Optimize performance
* Manage side effects (API calls, subscriptions)
* Handle updates and cleanups
* Reuse logic during specific phases

Implementing componentDidMount() hook and Implementing componentDidCatch() life cycle hook.

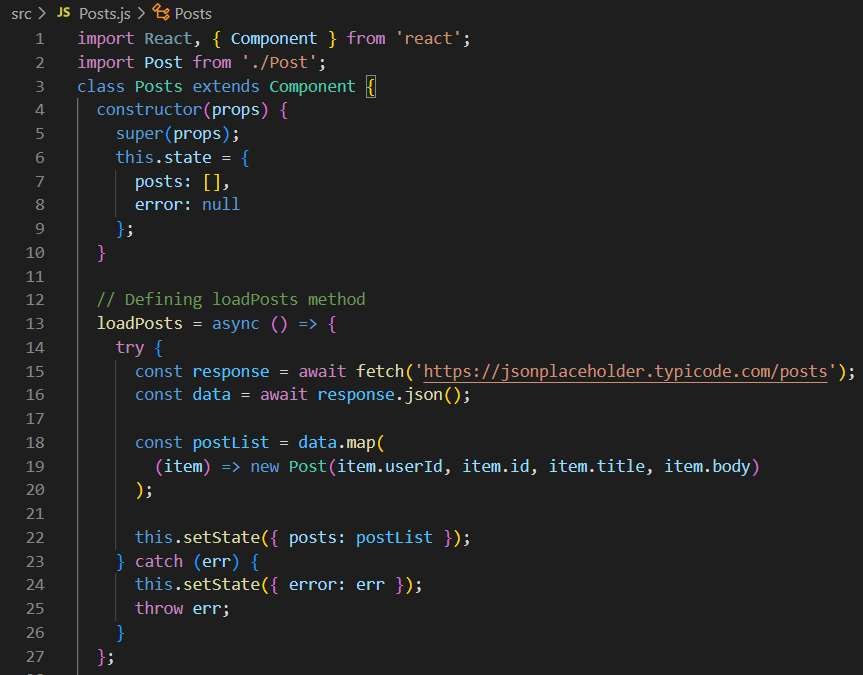
Creating a new react application using *create-react-app* tool with the name as “blogapp”

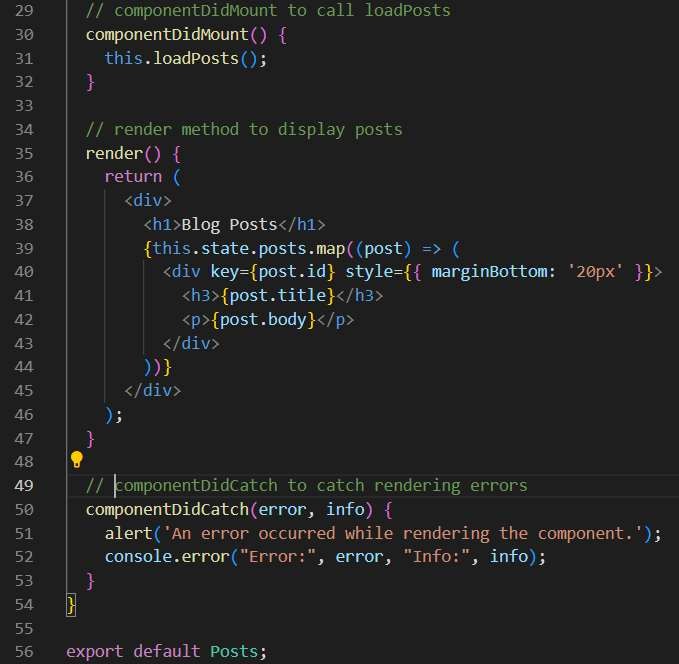


Post.js

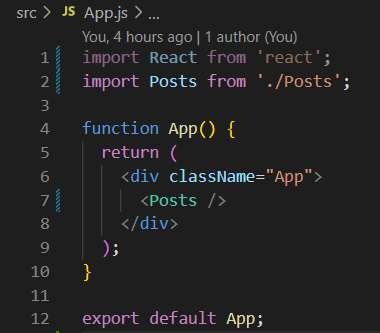


Posts.js

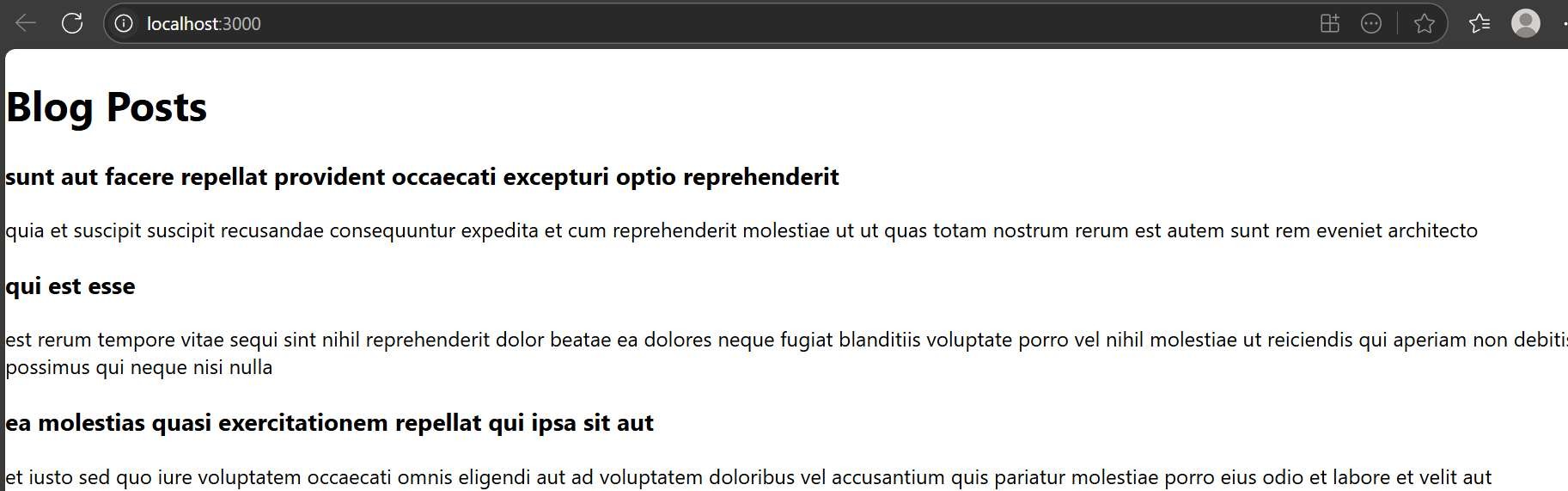




App.js

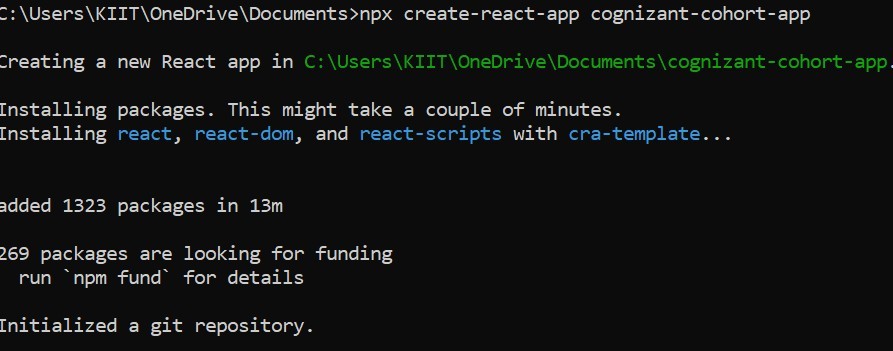


OUTPUT:



# MODULE 5:

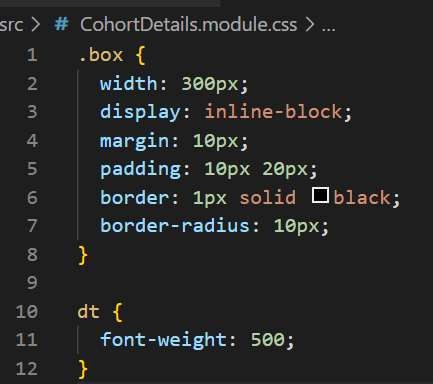
The Academy team at Cognizant wants to create a dashboard containing the details of ongoing and completed cohorts. A React application is created which displays the detail of the cohorts using Reactcomponent. You are assigned the task of styling these react components.



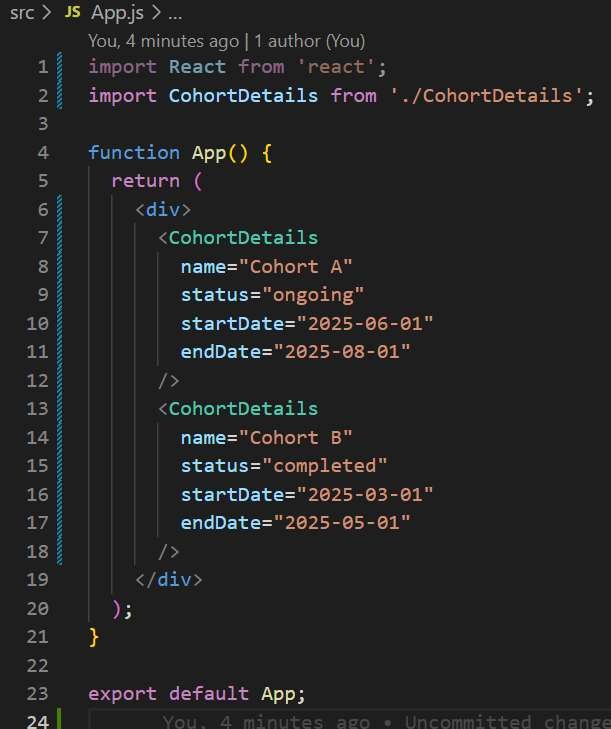
CohortDetails.js

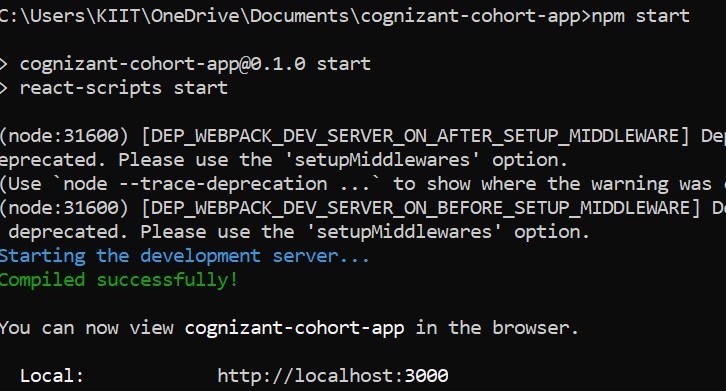


CohortDetails.module.css



App.js





OUTPUT:

