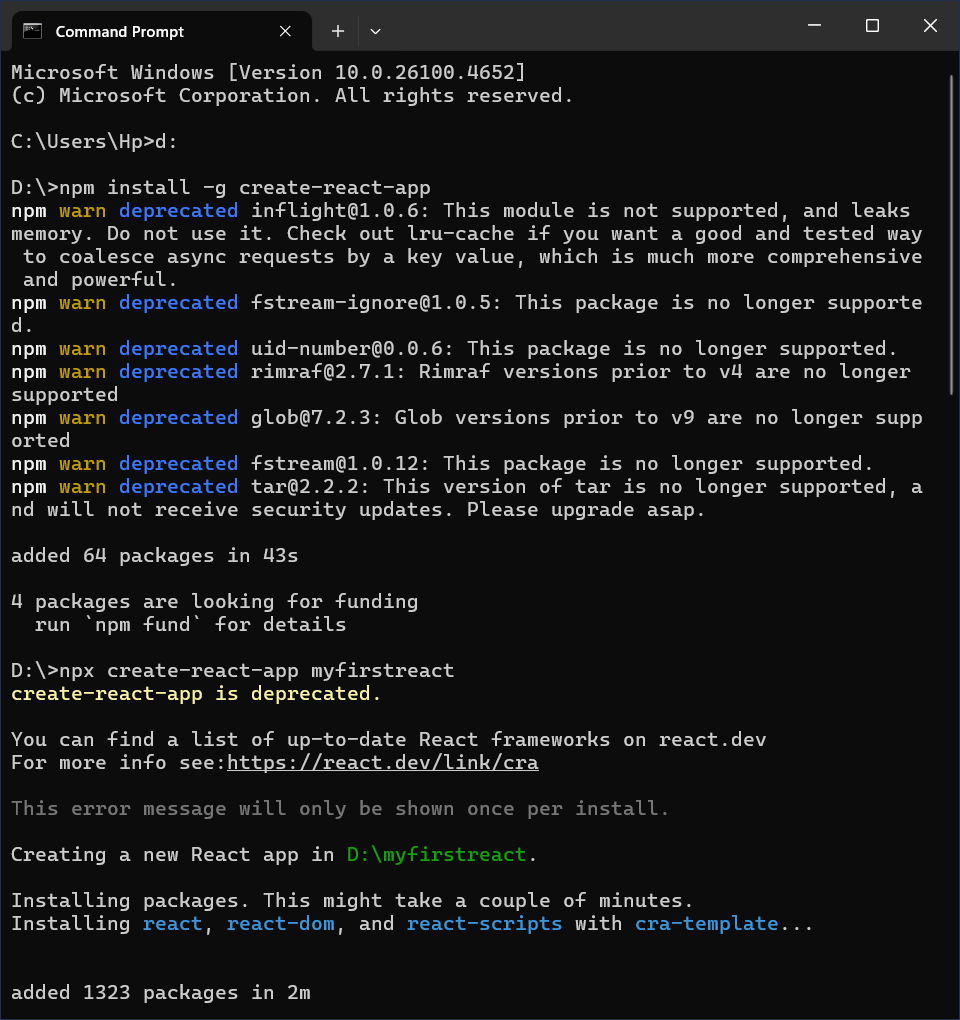
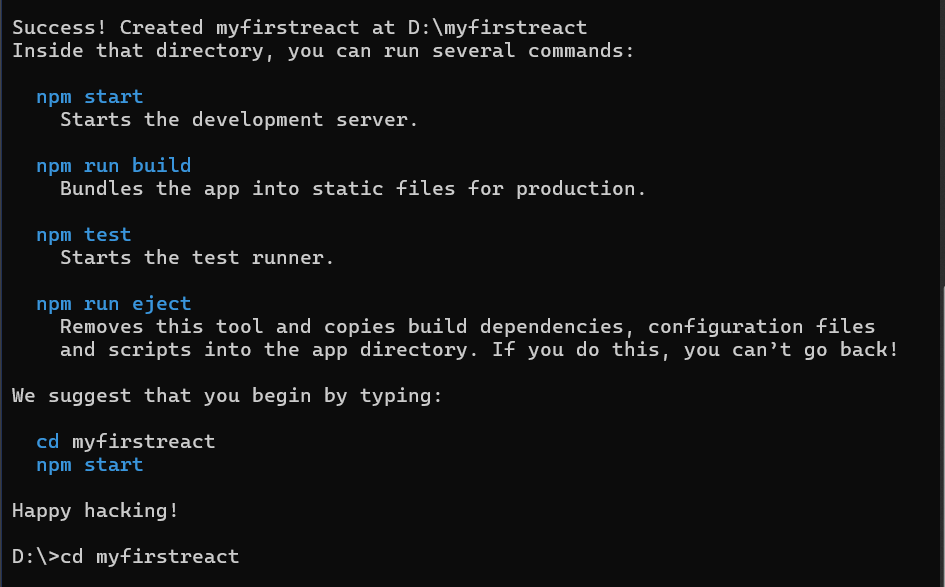
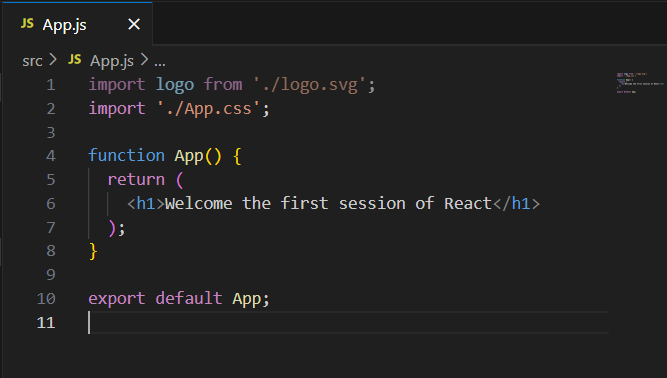
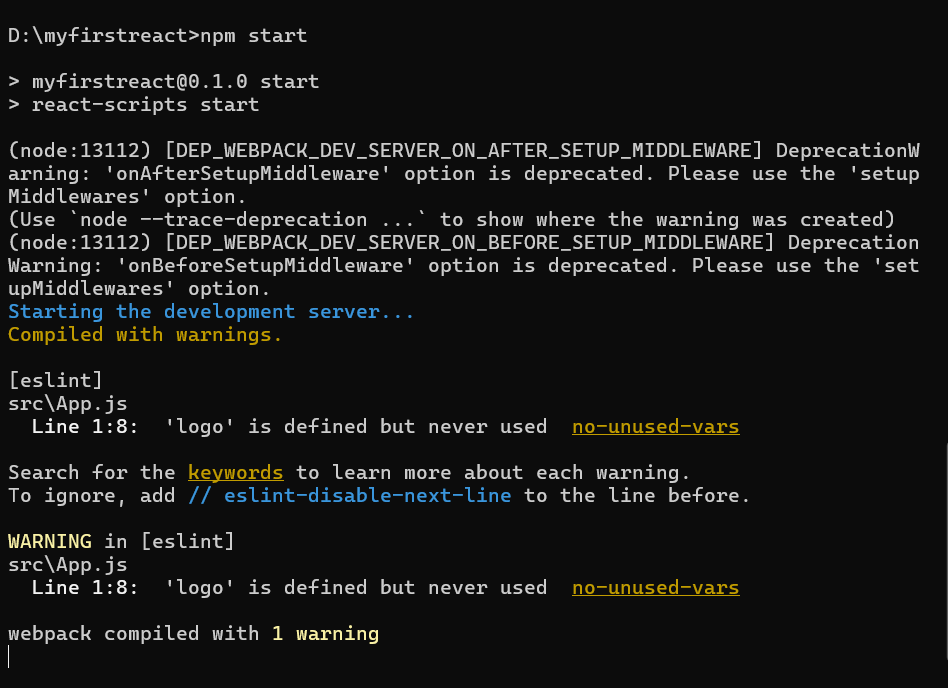
1. **ReactJS-Handson**

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.

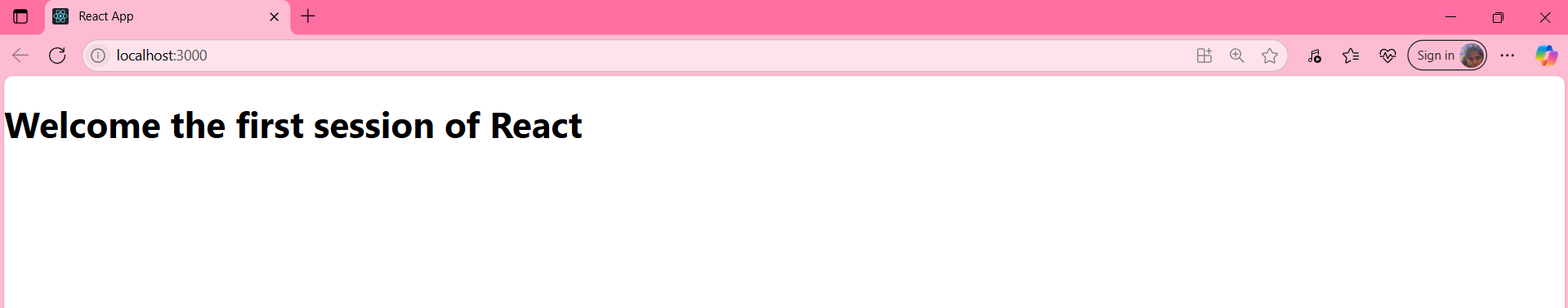








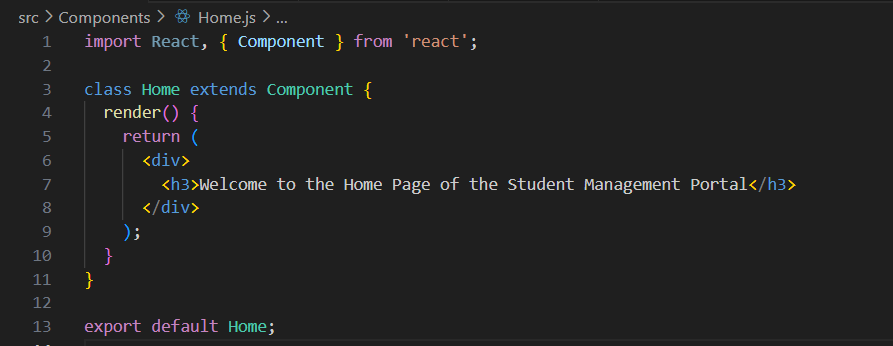
**OUTPUT:**



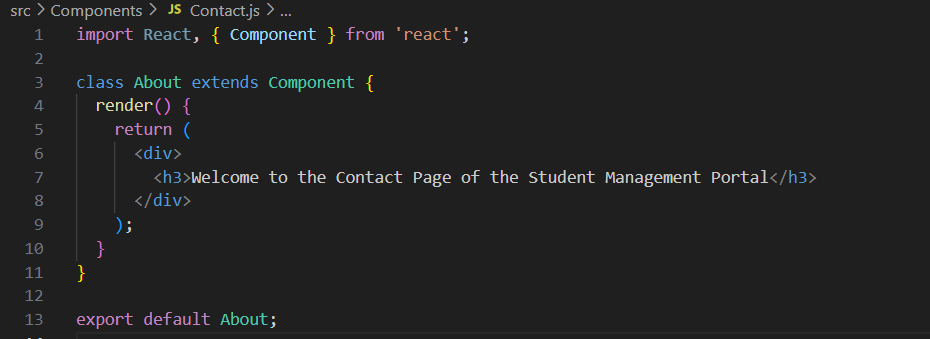
1. **ReactJS\_Hands\_On**

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.

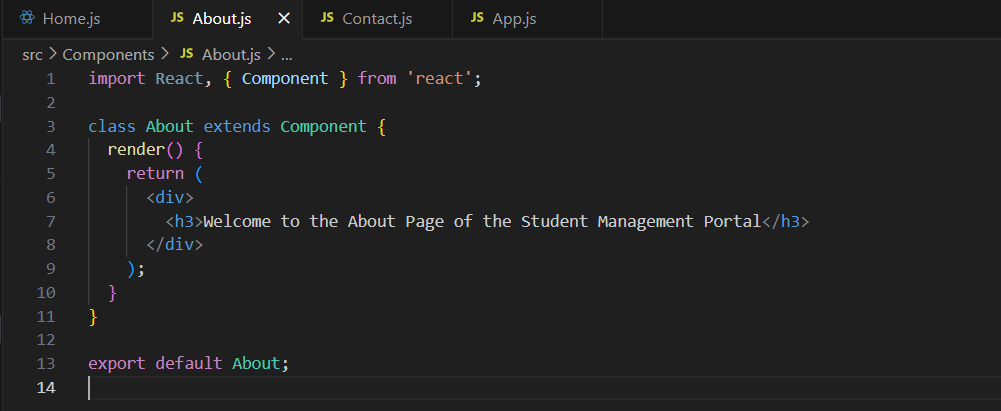
1. Create a new folder under Src folder with the name “Components”. Add a new file named “Home.js”



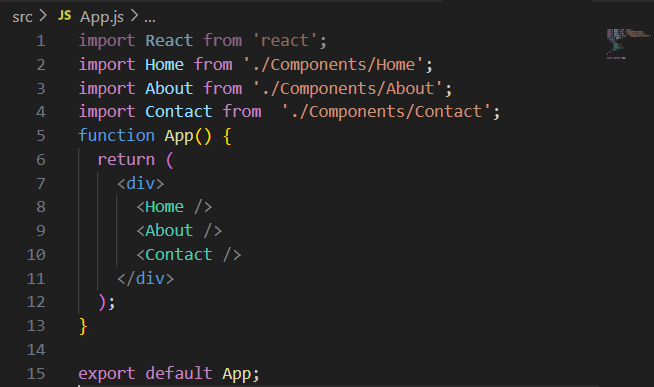
1. Add a new file named “Contact.js” in Components folder



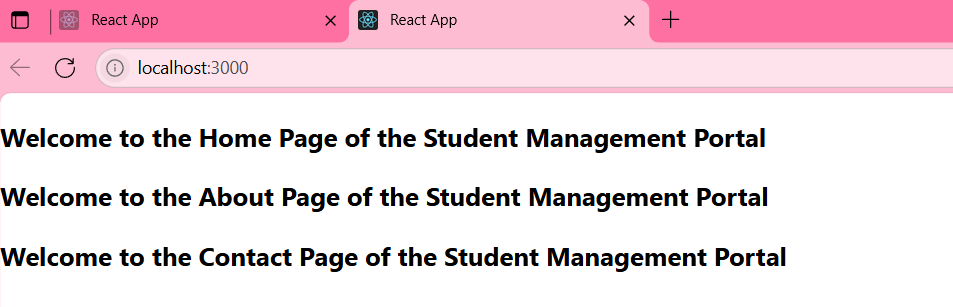
1. Add a new file named “About.js” in Components folder



1. Add a new file named “App.js” in src folder

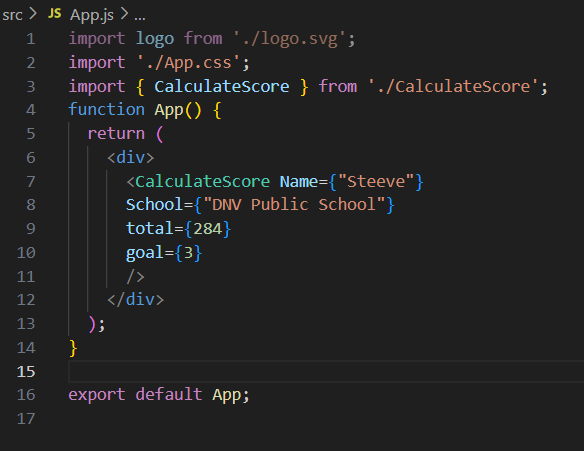


**OUTPUT:**

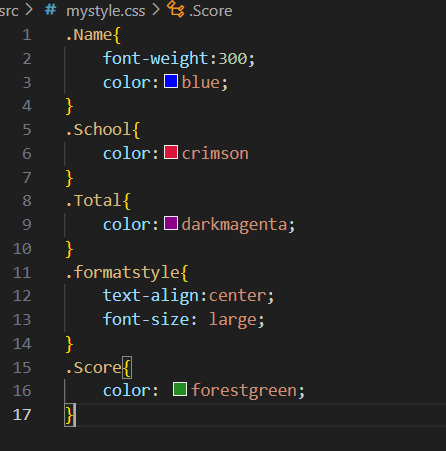


1. **ReactJS-Handson**

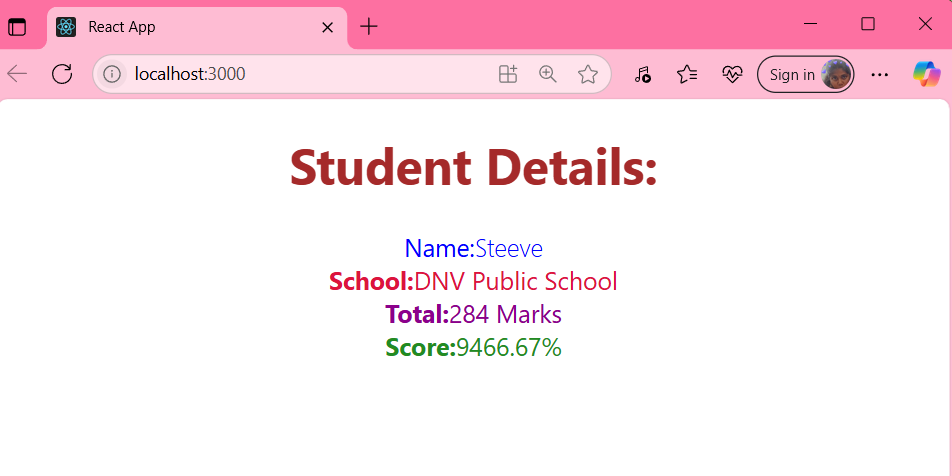
Create a react app for Student Management Portal named scorecalculatorapp and create a function component named “CalculateScore” which will accept Name, School, Total and goal in order to calculate the average score of a student and display the same.





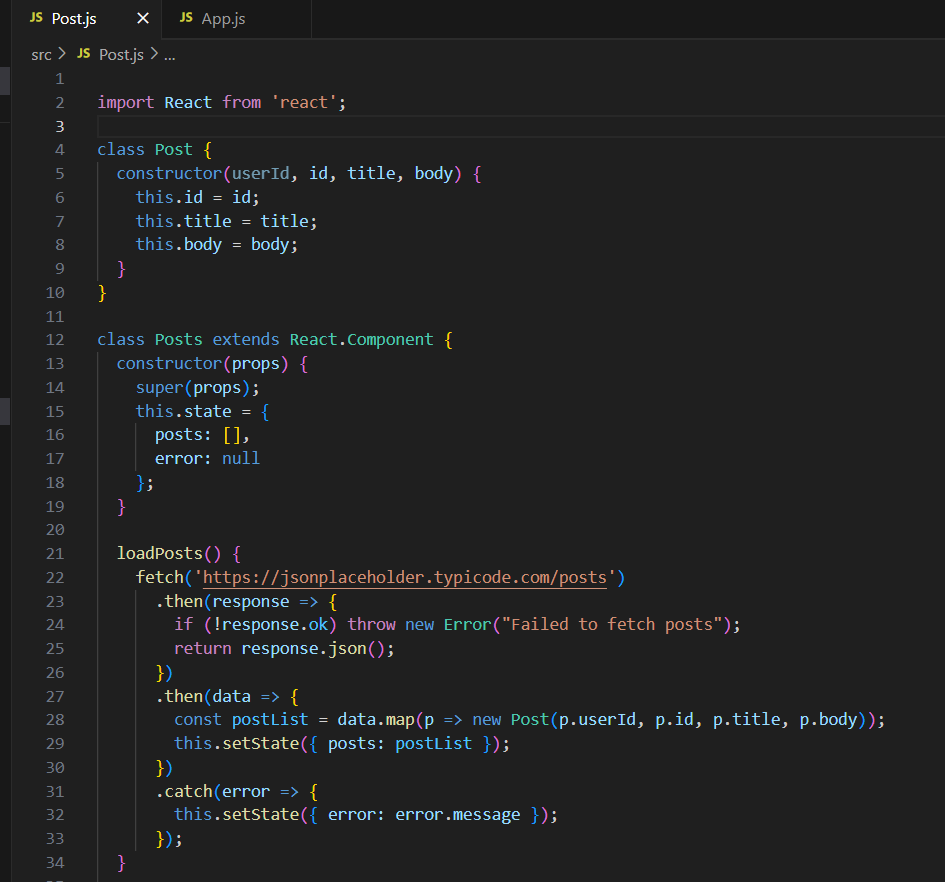


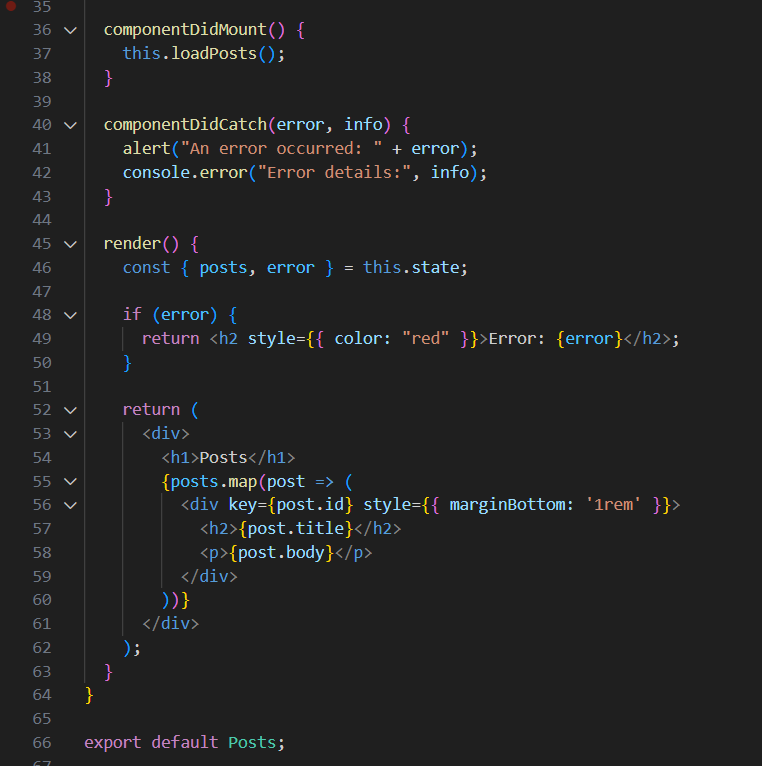
OUTPUT:



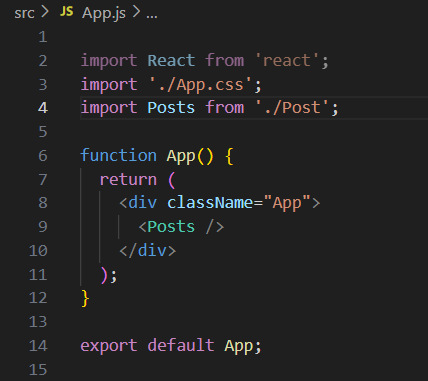
1. **ReactJS-Handson**
2. Create a new react application using *create-react-app* tool with the name as “blogapp”
3. Open the application using VS Code
4. Create a new file named as **Post.js** in **src folder** with following properties
5. Create a new class based component named as **Posts** inside **Posts.js** file
6. Initialize the component with a list of Post in state of the component using the constructor
7. 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>)
8. Implement the **componentDidMount()** hook to make calls to **loadPosts()** which will fetch the posts
9. Implement the **render()** which will display the title and post of posts in html page using heading and paragraphs respectively.
10. Define a **componentDidCatch()** method which will be responsible for displaying any error happing in the component as alert messages.

Posts.js

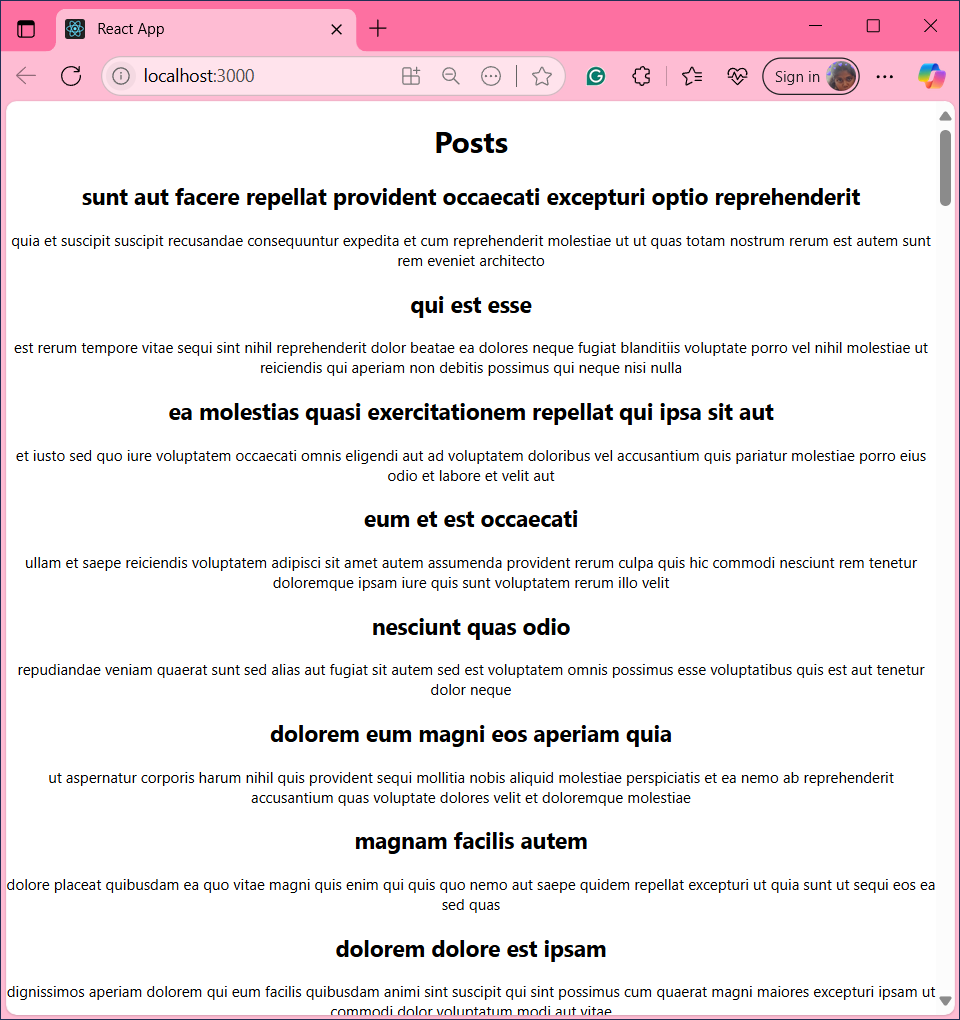


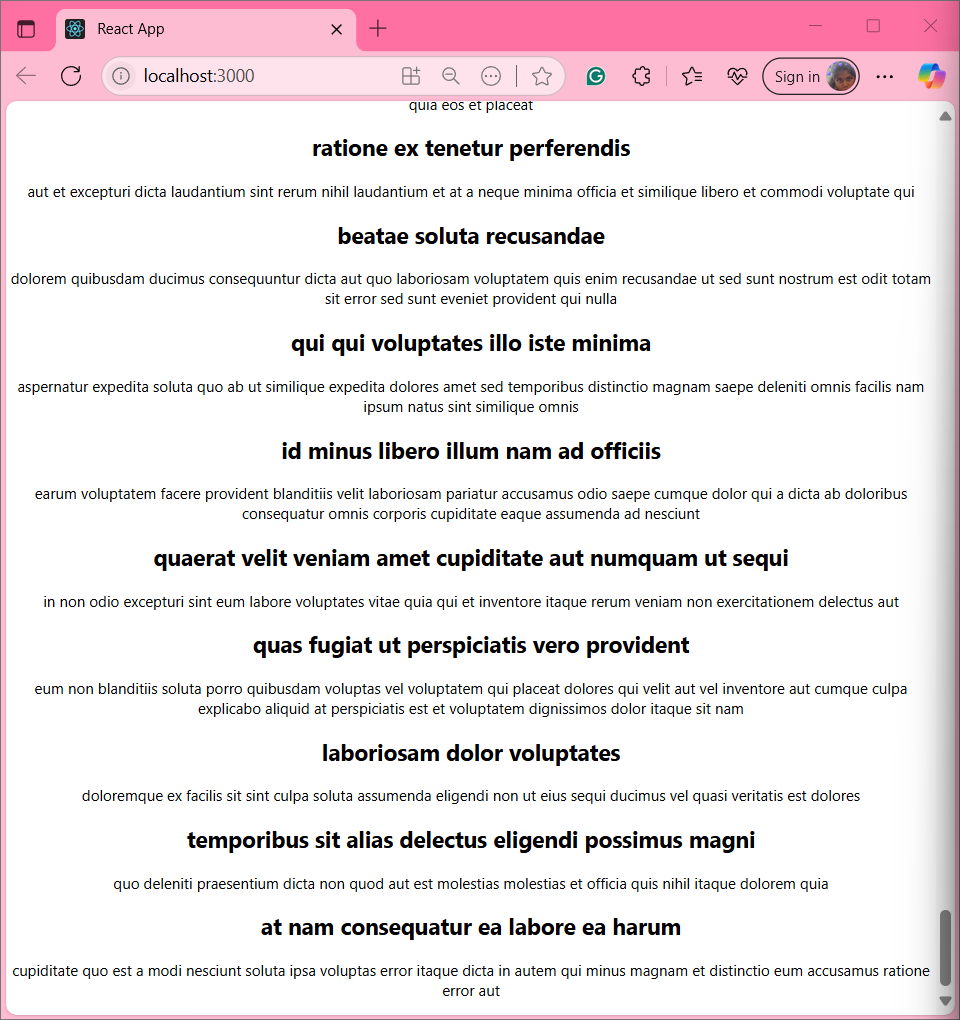


App.js



**OUTPUT:**

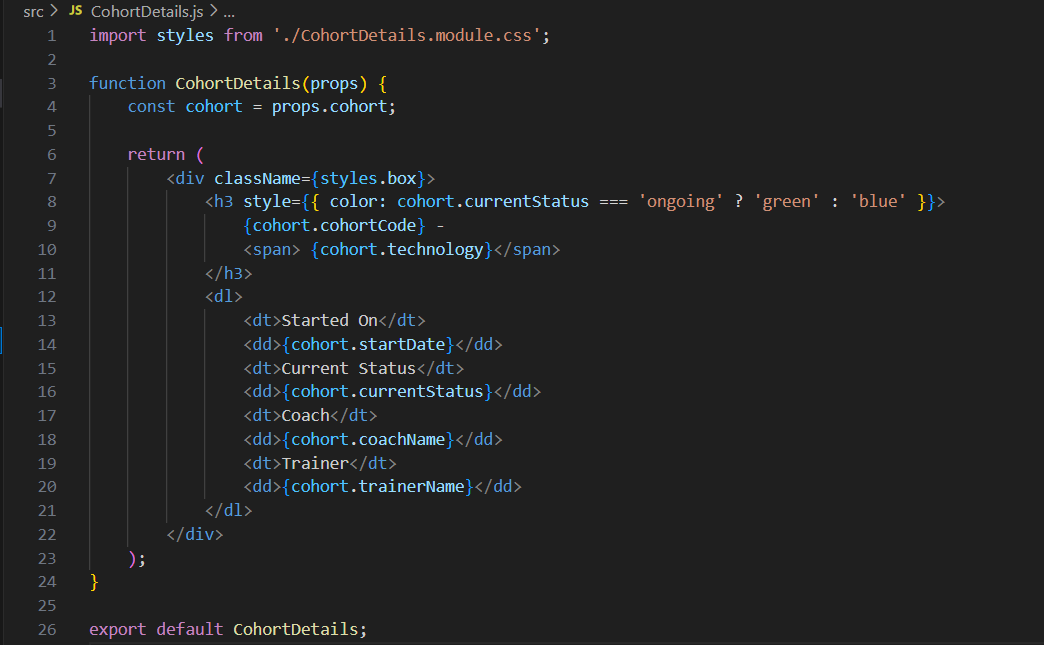




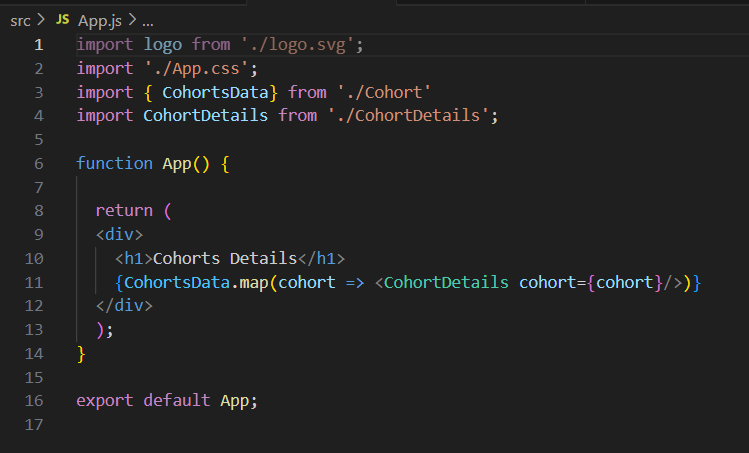
1. **ReactJS-Handson**

**TASK:** 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.

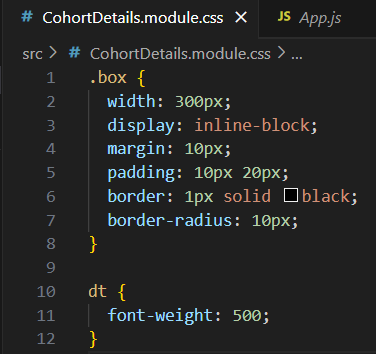
**CohortDetails.js**



**App.js**



**CohortDetails.module.css**



**OUTPUT:**

