**REACT JS**

Q1)

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.

### ****Step 1: Install Node.js and npm****

Go to: [https://nodejs.org/en/download](https://nodejs.org/en/download" \t "C:\\Users\\KIIT\\AppData\\Local\\Temp\\_new)

Download and install the **LTS version** for your OS.

After installation check version :-

node -v

npm -v

### ****Step 2: Install Create React App****

In your terminal or command prompt, run:

npm install -g create-react-app

### ****Step 3: Create the React App****

Run the following command:

npx create-react-app myfirstreact

### ****Step 4: Navigate to the App Folder****

cd myfirstreact

Step 5: Open the Project in VS Code

### ****Step 6: Edit**** App.js

import React from 'react';

function App() {

return (

<div>

<h1>welcome to the first session of React</h1>

</div>

);

}

export default App;

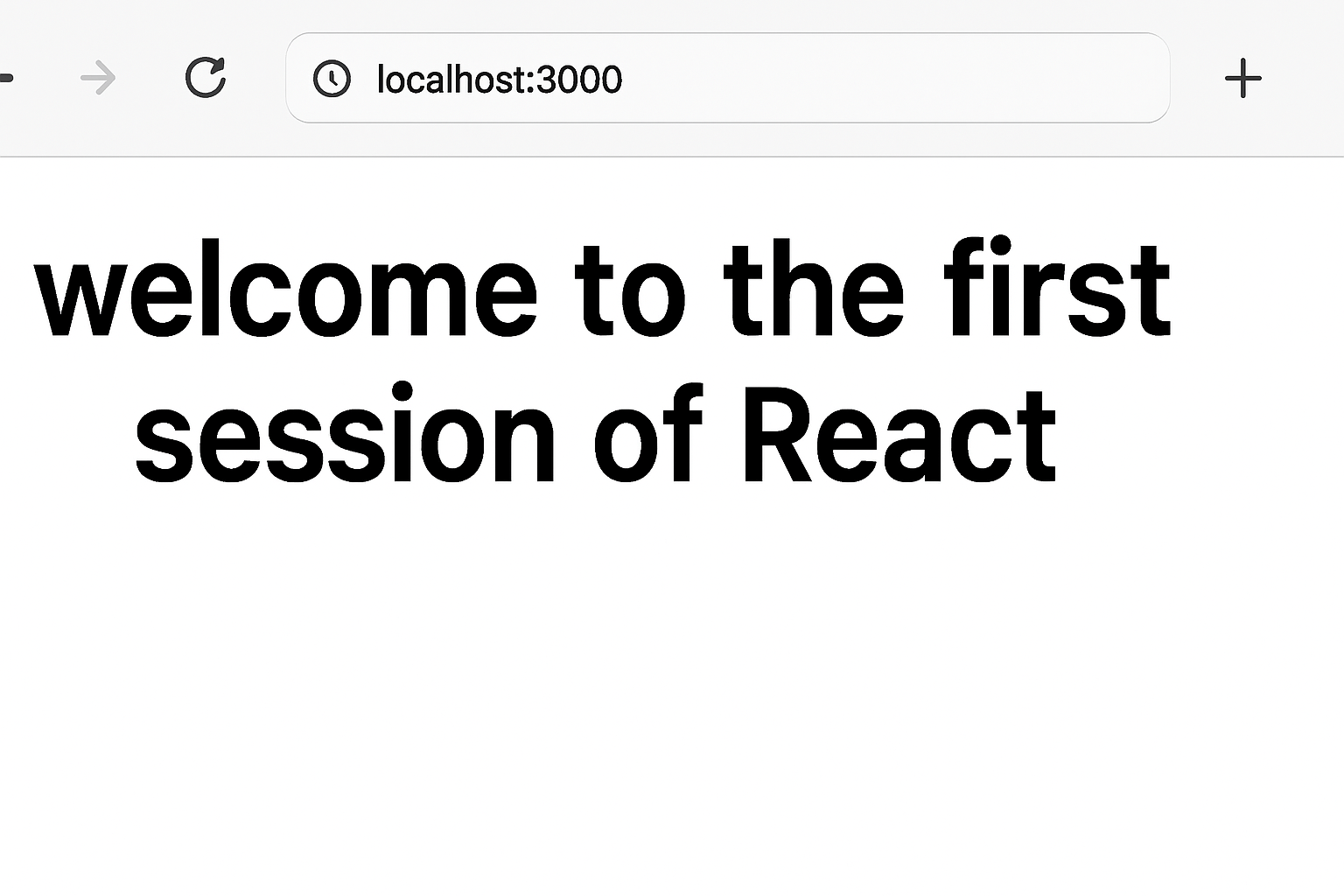
### ****Step 7: Run the React Application****

In the terminal inside VS Code, run:

npm start

### ****Step 8: View in Browser****

<http://localhost:3000>



Q2)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 React App named StudentApp****

In **Visual Studio Code** terminal (or any terminal), run:

npx create-react-app StudentApp

cd StudentApp

### ****2. Create Components Folder****

Inside src/, create a folder called Components.

### ****3. Create**** Home.js ****inside**** Components

src/Components/Home.js:

import React from 'react';

function Home() {

return (

<div>

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

</div>

);

}

export default Home;

**4. Create** About.js

src/Components/About.js:

import React from 'react';

function About() {

return (

<div>

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

</div>

);

}

export default About;

### ****5. Create**** Contact.js

src/Components/Contact.js:

import React from 'react';

function Contact() {

return (

<div>

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

</div>

);

}

export default Contact;

### ****6. Modify**** App.js ****to Call All Components****

src/App.js:

import React from 'react';

import Home from './Components/Home';

import About from './Components/About';

import Contact from './Components/Contact';

function App() {

return (

<div className="App">

<h1>Student Management Portal</h1>

<Home />

<About />

<Contact />

</div>

);

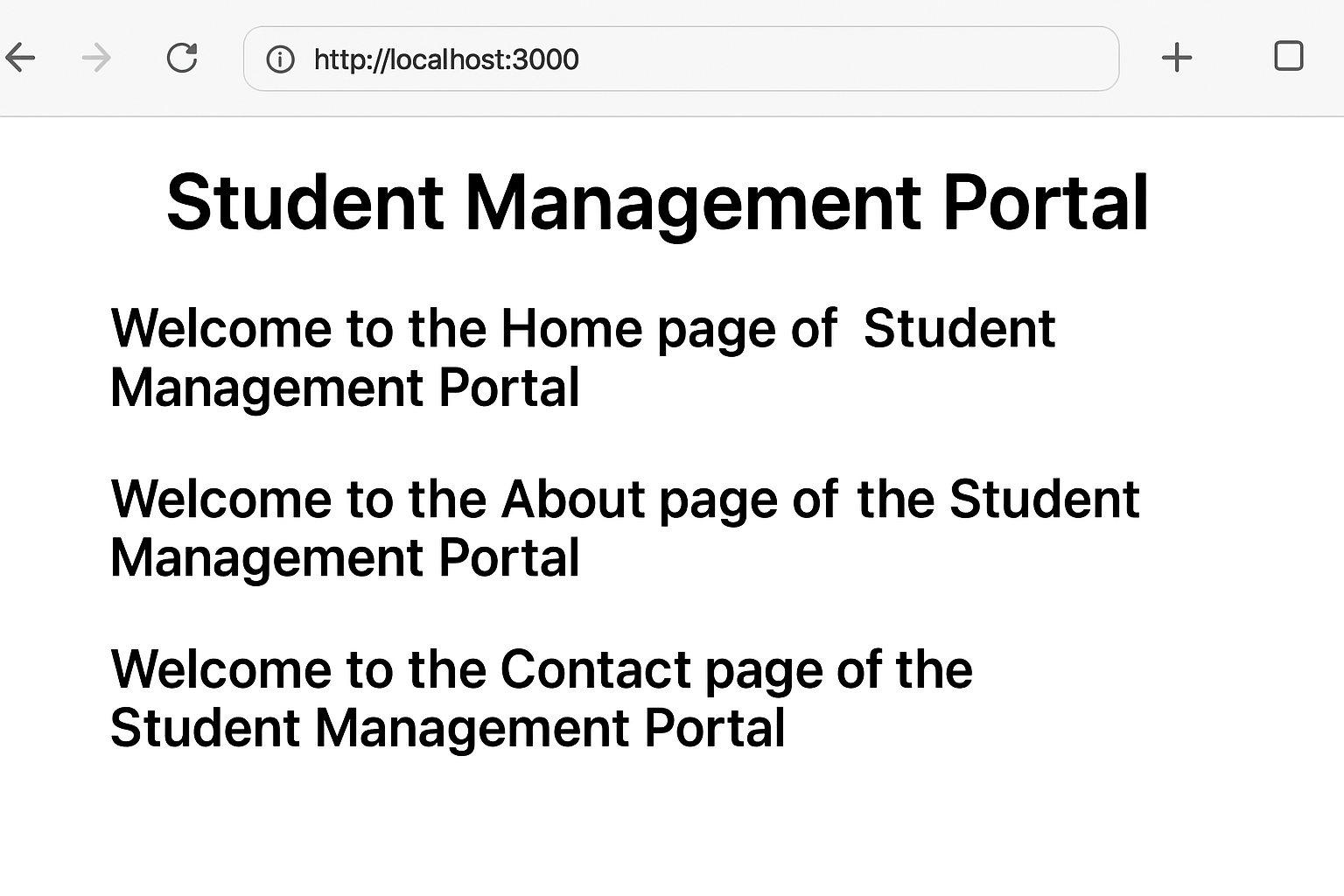
}

export default App;

**7. Run the App**

In terminal:

npm start



Q3)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.

Here’s the full implementation for your **Score Calculator App** based on the given steps:

### ****1. Create React App named**** scorecalculatorapp

In terminal:

npx create-react-app scorecalculatorapp

cd scorecalculatorapp

### ****2. Create Components Folder****

Inside src/, create a folder named Components and add a file CalculateScore.js.

### ****3. Code for**** CalculateScore.js

src/Components/CalculateScore.js:

import React, { useState } from 'react';

import '../Stylesheets/mystyle.css';

function CalculateScore() {

const [name, setName] = useState('');

const [school, setSchool] = useState('');

const [total, setTotal] = useState('');

const [goal, setGoal] = useState('');

const [average, setAverage] = useState(null);

const calculateAverage = () => {

if (total && goal) {

const avg = (parseFloat(total) / parseFloat(goal)).toFixed(2);

setAverage(avg);

}

};

return (

<div className="container">

<h2>Student Score Calculator</h2>

<input

type="text"

placeholder="Enter Student Name"

value={name}

onChange={(e) => setName(e.target.value)}

/>

<input

type="text"

placeholder="Enter School Name"

value={school}

onChange={(e) => setSchool(e.target.value)}

/>

<input

type="number"

placeholder="Enter Total Score"

value={total}

onChange={(e) => setTotal(e.target.value)}

/>

<input

type="number"

placeholder="Enter Goal Score"

value={goal}

onChange={(e) => setGoal(e.target.value)}

/>

<button onClick={calculateAverage}>Calculate Average</button>

{average && (

<div className="result">

<h3>Student Details:</h3>

<p>Name: {name}</p>

<p>School: {school}</p>

<p>Total: {total}</p>

<p>Goal: {goal}</p>

<p>Average Score: {average}</p>

</div>

)}

</div>

);

}

export default CalculateScore;

### ****4. Create Stylesheet****

Create folder src/Stylesheets and add mystyle.css:

src/Stylesheets/mystyle.css:

.container {

width: 400px;

margin: 50px auto;

padding: 20px;

border: 2px solid #333;

border-radius: 10px;

background-color: #f9f9f9;

text-align: center;

font-family: Arial, sans-serif;

}

input {

display: block;

width: 90%;

padding: 8px;

margin: 10px auto;

border-radius: 5px;

border: 1px solid #ccc;

}

button {

padding: 10px 20px;

background-color: #007bff;

color: white;

border: none;

border-radius: 5px;

cursor: pointer;

}

button:hover {

background-color: #0056b3;

}

.result {

margin-top: 20px;

text-align: left;

}

### ****5. Edit**** App.js ****to Invoke Component****

Replace src/App.js with:

import React from 'react';

import CalculateScore from './Components/CalculateScore';

function App() {

return (

<div className="App">

<h1>Score Calculator App</h1>

<CalculateScore />

</div>

);

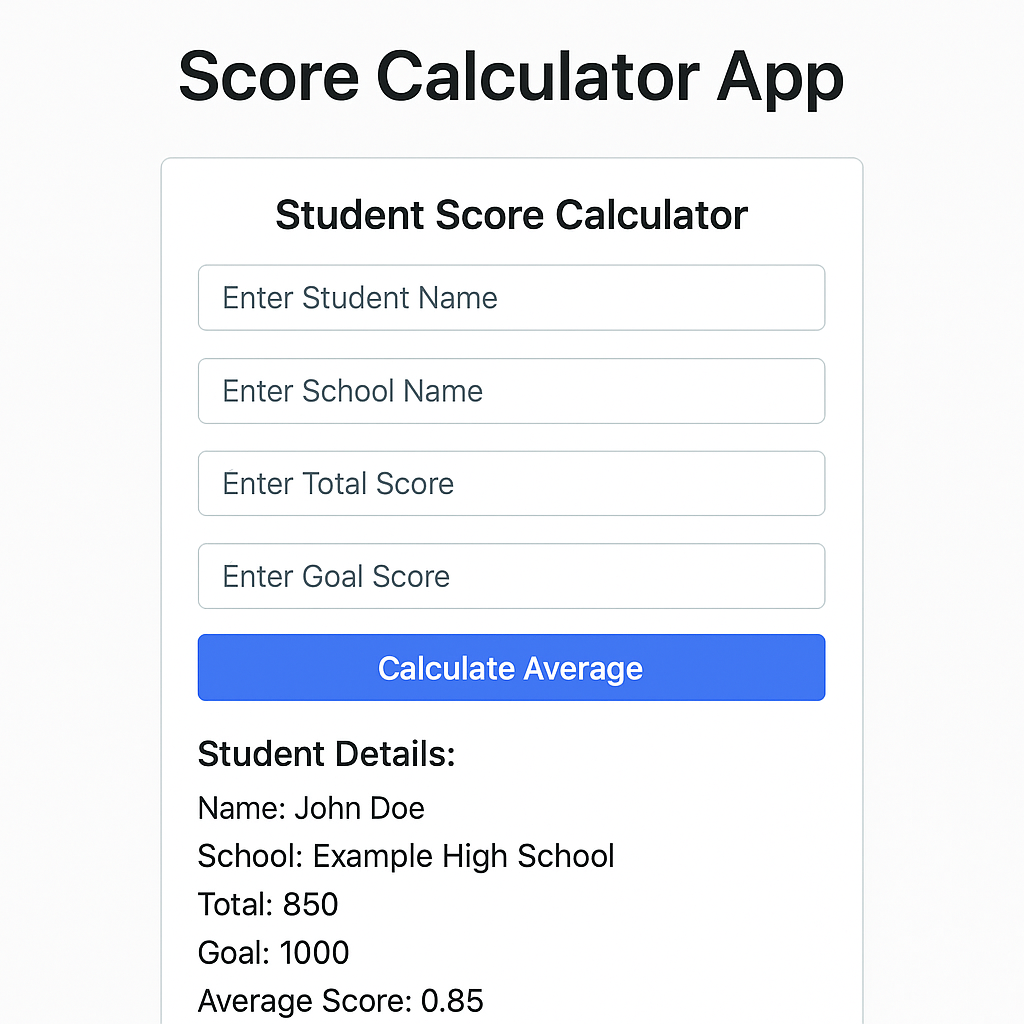
}

export default App;

### ****6. Run the App****

In terminal:

npm start



Q4)Create a new react application using *create-react-app* tool with the name as “blogapp”

### ****1. Create a React Application****

Open the terminal and run:

npx create-react-app blogapp

cd blogapp

### ****2. Open in VS Code****

Open the folder **blogapp** in Visual Studio Code:

code .

### ****3. Create**** Post.js ****in**** src ****Folder****

This will represent a single post entity as a class:

src/Post.js:

class Post {

constructor(id, title, body) {

this.id = id;

this.title = title;

this.body = body;

}

}

export default Post;

**4. Create** Posts.js **Class-Based Component**

Inside src/, create Posts.js:

import React, { Component } from 'react';

import Post from './Post';

class Posts extends Component {

constructor(props) {

super(props);

this.state = {

posts: [],

error: null

};

}

// 6. loadPosts method using Fetch API

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 });

});

}

// 7. componentDidMount hook to load posts

componentDidMount() {

this.loadPosts();

}

// 9. componentDidCatch for error handling

componentDidCatch(error, info) {

alert('Error occurred: ' + error);

}

// 8. render posts in HTML

render() {

const { posts } = this.state;

return (

<div>

<h1>Blog Posts</h1>

{posts.length > 0 ? (

posts.map(post => (

<div key={post.id}>

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

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

<hr />

</div>

))

) : (

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

)}

</div>

);

}

}

export default Posts;

### ****10. Add Posts Component to App.js****

src/App.js:

import React from 'react';

import Posts from './Posts';

function App() {

return (

<div className="App">

<Posts />

</div>

);

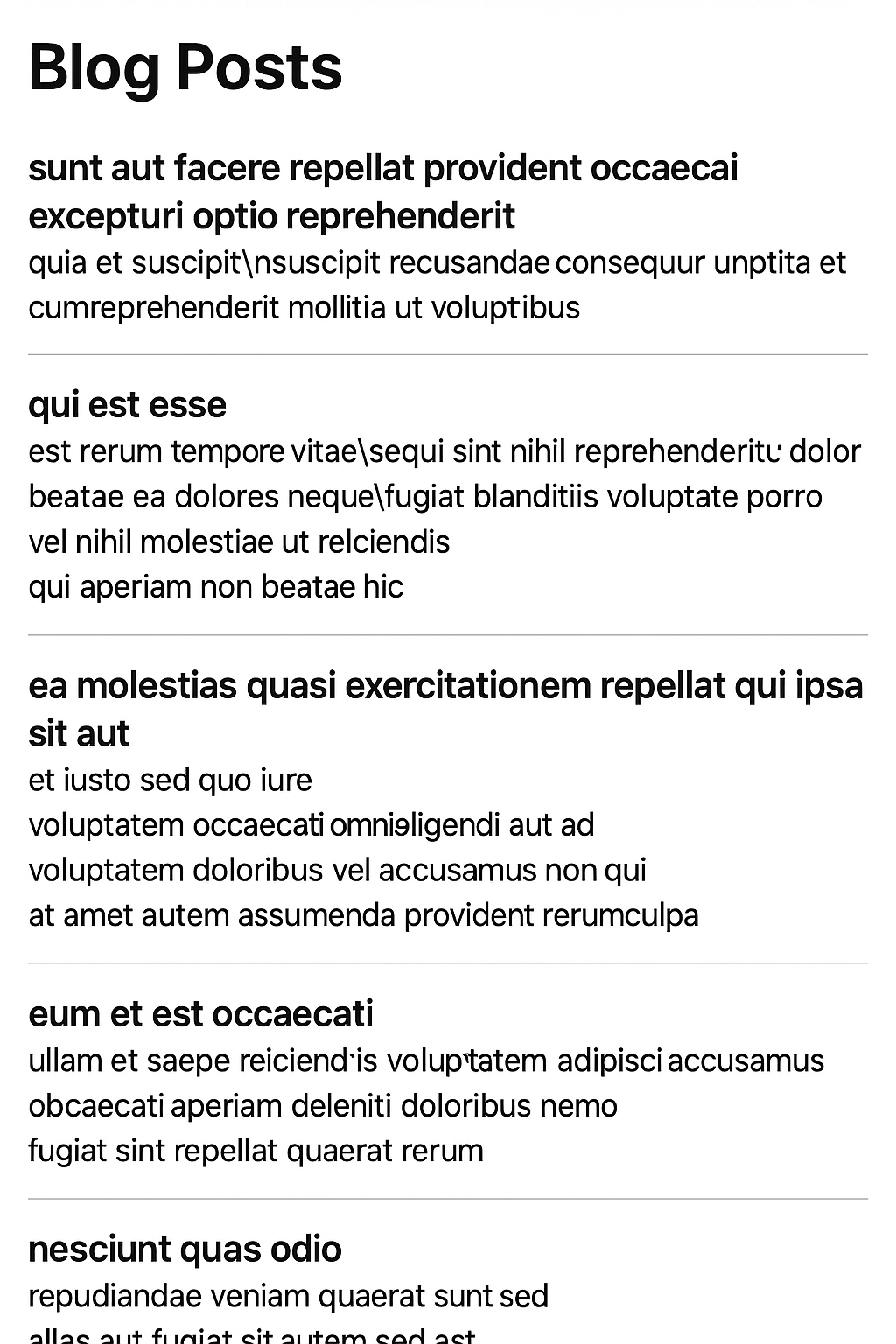
}

export default App;

### ****11. Run the Application****

In the terminal:

npm start



Q5)

### ****1. Unzip & Open Application****

Unzip the provided React app into a folder.

Open a **Command Prompt** or **terminal**, navigate to the project folder:

cd your-app-folder

### ****2. Restore Node Packages****

Run:

npm install

### ****3. Open in VS Code****

Open the app:

code .

### ****4. Create CSS Module**** CohortDetails.module.css

In src/ (or inside the components folder where CohortDetails exists), create:

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;

}

### ****5. Open**** CohortDetails.js ****Component and Import CSS****

Inside CohortDetails.js:

import React from 'react';

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

function CohortDetails({ name, status, startDate, endDate }) {

const headingColor = status === 'ongoing' ? 'green' : 'blue';

return (

<div className={styles.box}>

<h3 style={{ color: headingColor }}>{name}</h3>

<dl>

<dt>Status:</dt>

<dd>{status}</dd>

<dt>Start Date:</dt>

<dd>{startDate}</dd>

<dt>End Date:</dt>

<dd>{endDate}</dd>

</dl>

</div>

);

}

export default CohortDetails;

### ****6. Apply**** box ****class****

The div wrapping the content now uses:

<div className={styles.box}>

This applies the styles from **CSS module**.

### ****7. Dynamic Heading Color****

Condition:

const headingColor = status === 'ongoing' ? 'green' : 'blue';

Applied in:

<h3 style={{ color: headingColor }}>{name}</h3>

### ****8. Run the Application****

npm start

