**Hands-On Exercise: 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.

App.js

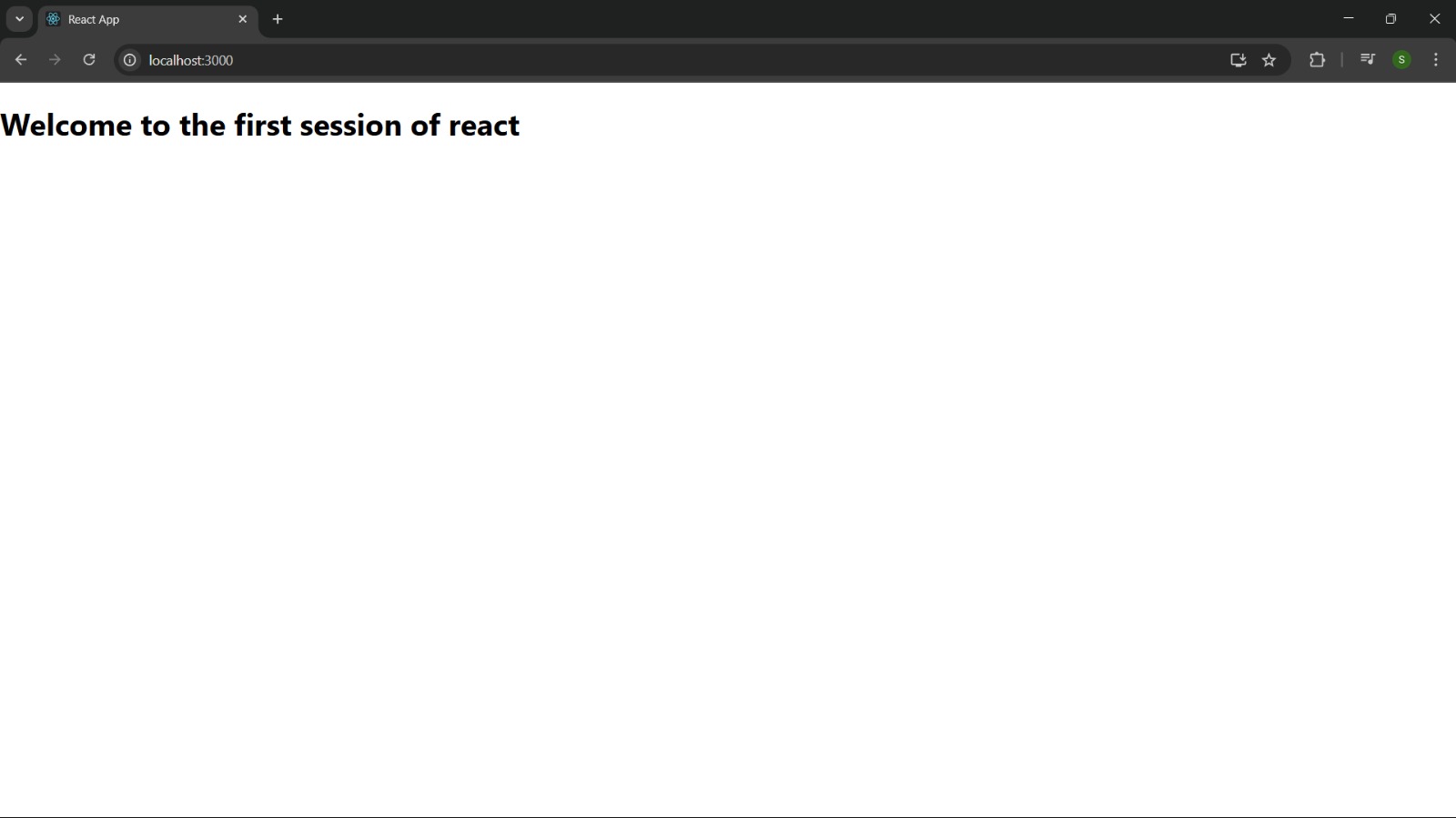
function App() {

return (

<h1>Welcome to the first session of react</h1>

);

}

export default App;  
  
  


**Hands-On Exercise: 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.

App.js  
  
import Home from './Components/Home';

import About from './Components/About';

import Contact from './Components/Contact';

function App() {

return (

<div className="container">

<Home />

<About />

<Contact />

</div>

);

}

export default App;

import React, { Component } from 'react';

class Home extends Component {

render() {

return (

<div>

<h3>Welcome to the Home Page of Student Management Portal</h3>

</div>

);

}

}

export default Home;

import React, { Component } from 'react';

class About extends Component {

render() {

return (

<div>

<h3>Welcome to the About Page of Student Management Portal</h3>

</div>

);

}

}

export default About;

Home.js

import React, { Component } from 'react';

class Contact extends Component {

render() {

return (

<div>

<h3>Welcome to the Contact Page of Student Management Portal</h3>

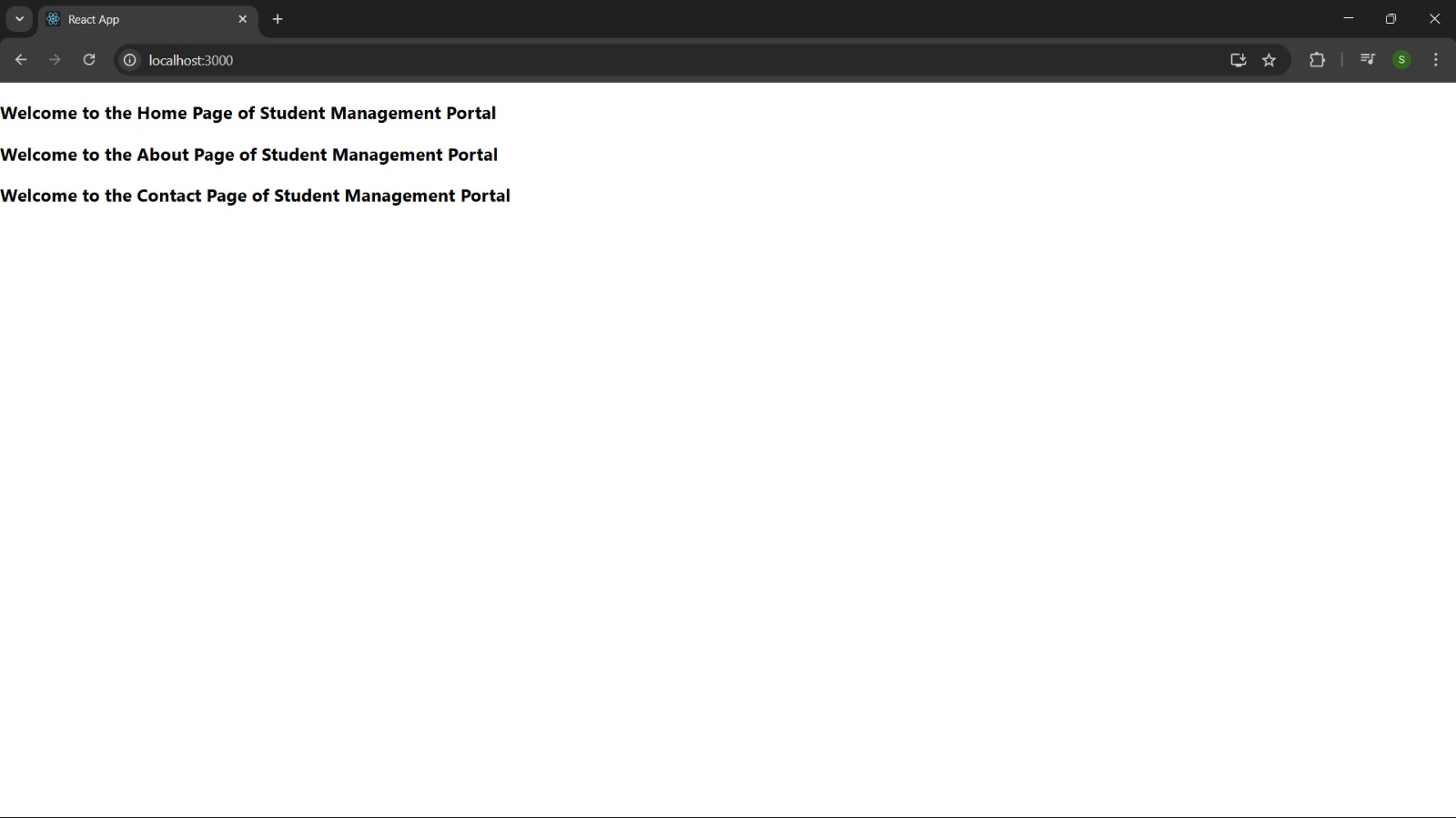
</div>

);

}

}

export default Contact;



**Hands-On Exercise: 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.

CalculateScore.js

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

function toDecimal(decimal) {

return decimal \* 100;

}

function percent(decimal) {

return toDecimal(decimal).toFixed(2) + '%';

}

function calcScore(total, goal) {

if (!goal || goal === 0) return 'N/A';

return percent(total / goal);

}

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

<div className="formatstyle">

<h1 style={{ color: 'brown' }}>Student Details:</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><span>Score:</span></b>

<span> {calcScore(total, goal)}</span>

</div>

</div>

);  
  
  
mystyle.css  
  
.Name {

font-weight: 300;

color: blue;

}

.School {

color: crimson;

}

.Total {

color: darkmagenta;

}

.Score {

color: forestgreen;

font-size: large;

}

.formatstyle {

text-align: center;

}  
  
  
  
App.js  
  
import { CalculateScore } from "./Components/CalculateScore";

function App() {

return (

<div>

<CalculateScore

Name="Sleeve"

School="DNV Public School"

total={284}

goal={3}

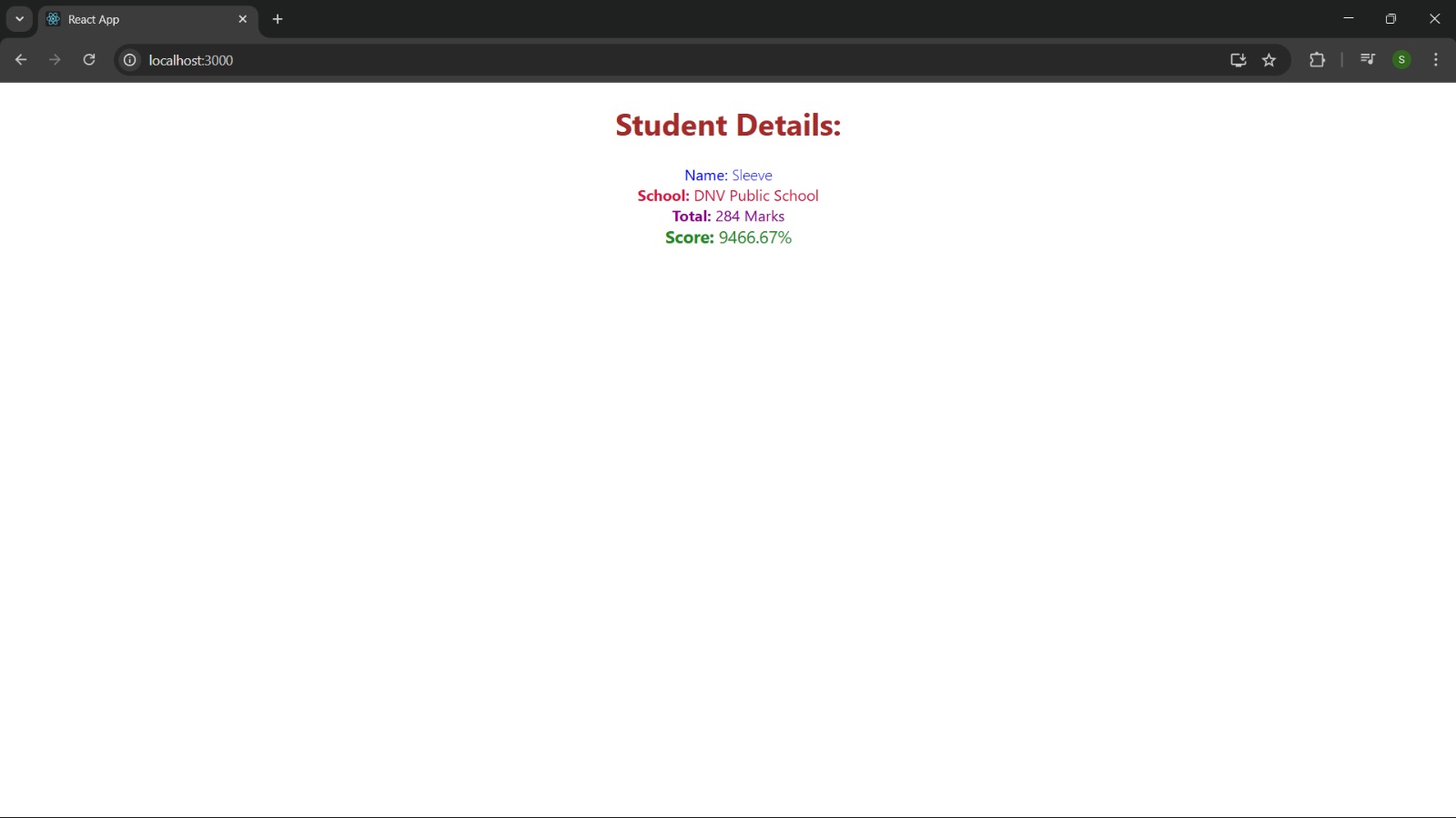
/>

</div>

);

}

export default App;



**Hands-On Exercise: 4**

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

export default class Post {

constructor(userId, id, title, body) {

this.userId = userId;

this.id = id;

this.title = title;

this.body = body;

}

}

import React, { Component } from 'react';

import Post from './Post';

export default class Posts extends Component {

constructor(props) {

super(props);

this.state = {

posts: []

};

}

loadPosts() {

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

.then(response => {

if (!response.ok) {

throw new Error('Network response was not ok');

}

return response.json();

})

.then(data => {

this.setState({ posts: data });

})

.catch(error => {

throw error;

});

}

componentDidMount() {

this.loadPosts();

}

componentDidCatch(error, info) {

alert(error.message);

}

render() {

return (

<div>

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

<div key={post.id}>

<h2>{post.title}</h2>

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

</div>

))}

</div>

);

}

}

import React from 'react';

import Posts from './Posts';

function App() {

return (

<div className="App">

<h1>Blog Posts</h1>

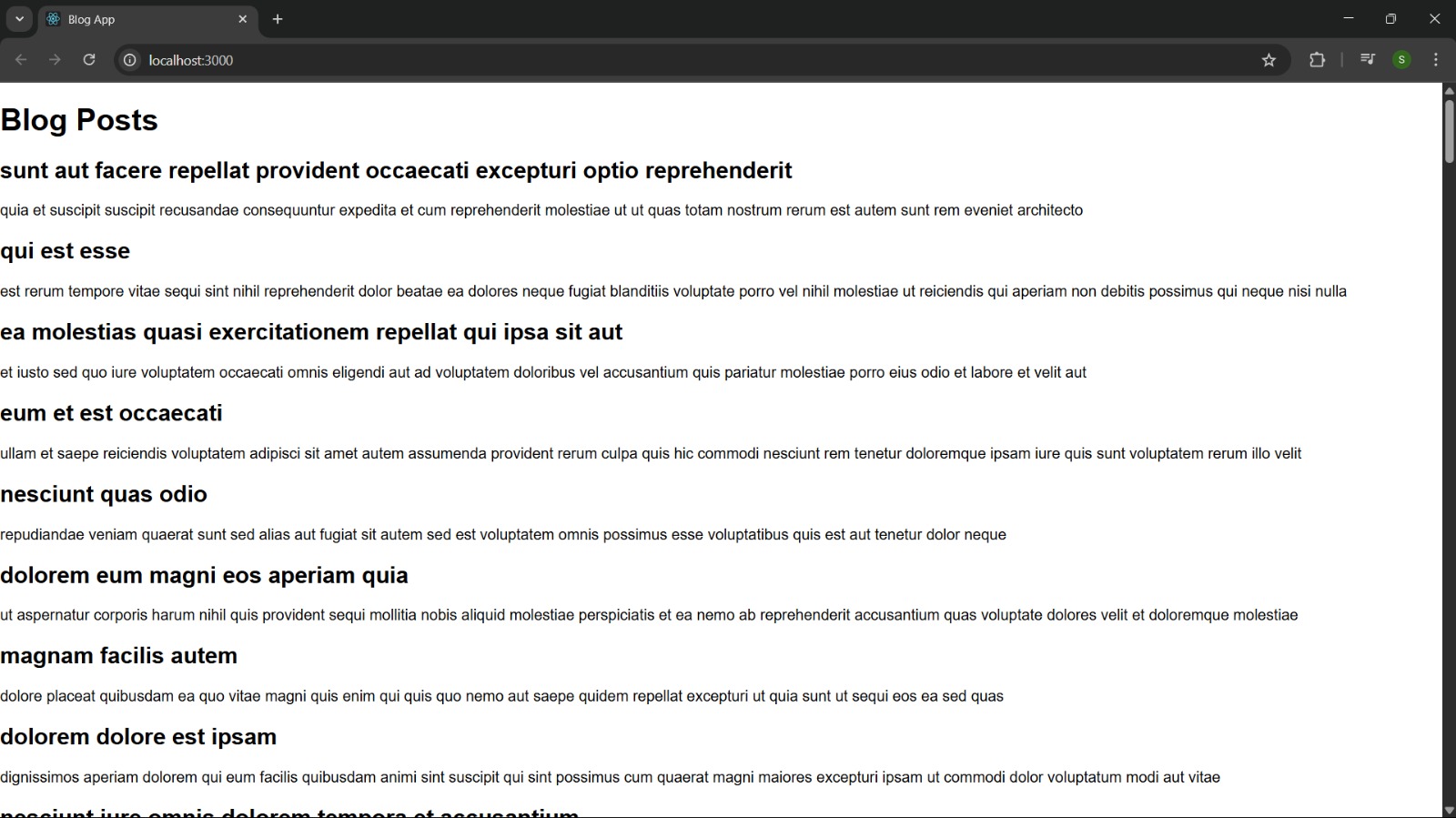
<Posts />

</div>

);

}

export default App;



**Hands-On Exercise: 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.

Download and build the attached react application.

import React from 'react';

import CohortDetails from './CohortDetails';

function App() {

return (

<div className="App">

<h1>Cohort Details</h1>

<CohortDetails />

</div>

);

}

export default App;

import React from 'react';

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

const cohorts = [

{ id: 1, name: 'Cohort A', status: 'ongoing', startDate: '2025-01-01', endDate: '2025-06-30' },

{ id: 2, name: 'Cohort B', status: 'completed', startDate: '2024-01-01', endDate: '2024-06-30' }

];

export default function CohortDetails() {

return (

<div>

{cohorts.map(cohort => (

<div key={cohort.id} className={styles.box}>

<h3 style={{ color: cohort.status === 'ongoing' ? 'green' : 'blue' }}>

{cohort.name}

</h3>

<dl>

<dt>Status</dt>

<dd>{cohort.status}</dd>

<dt>Start Date</dt>

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

<dt>End Date</dt>

<dd>{cohort.endDate}</dd>

</dl>

</div>

))}

</div>

);

}

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;

}

