**ReactJS HOL 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.

import logo from './logo.svg';

import './App.css';

function App() {

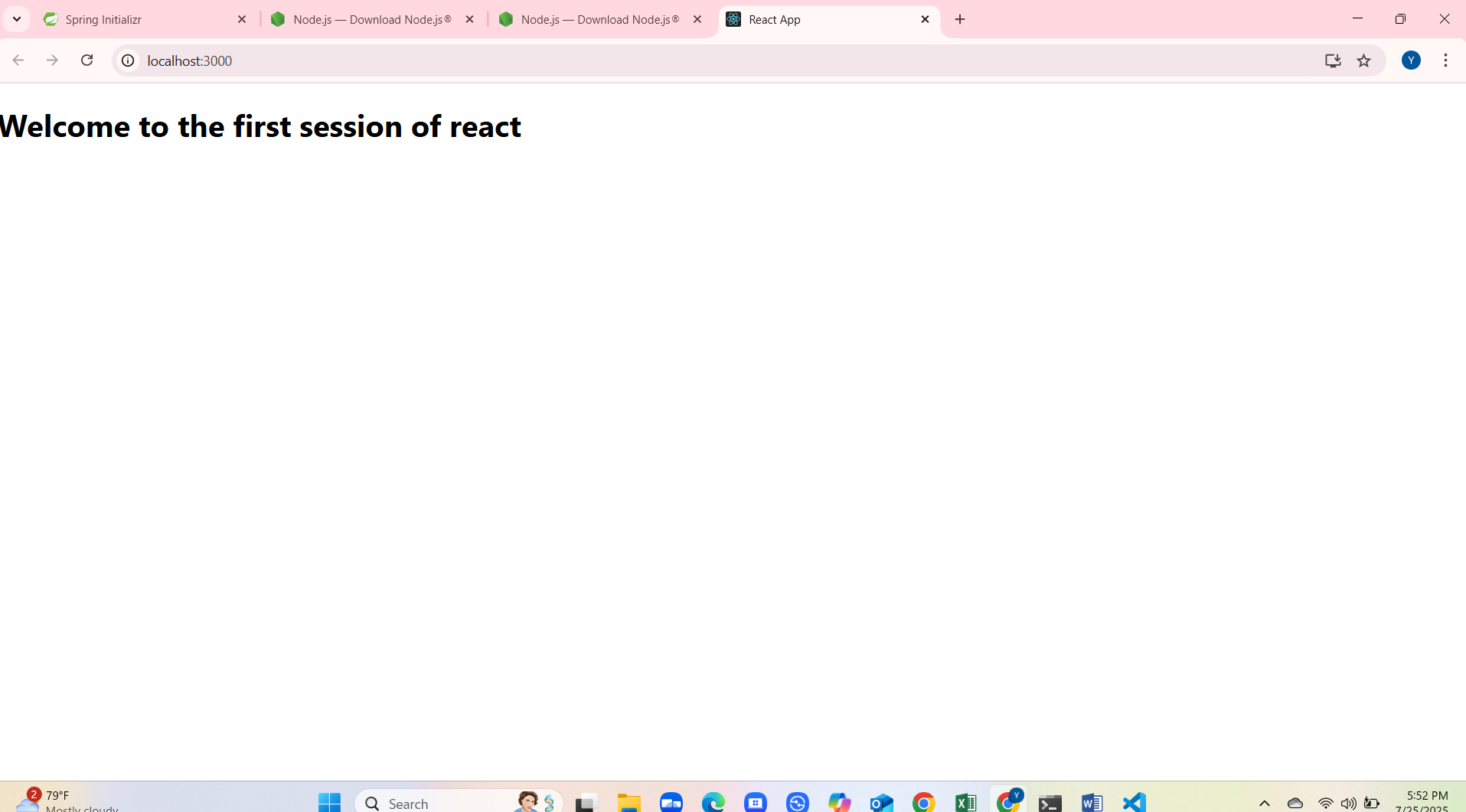
return(

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

);

}

export default App;



**Reactjs HOL 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

**Home.js**

import React,{Component} from "react";

export class Home extends Component{

render(){

return(

<div>

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

</div>

);

}

}

**About.js**

import React,{Component} from "react";

export class About extends Component{

render(){

return(

<div>

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

</div>

);

}

}

**Contact.js**

import React,{Component} from "react";

export class Contact extends Component{

render(){

return(

<div>

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

</div>

);

}

}

**App.js**

import logo from './logo.svg';

import './App.css';

import {Home} from './Components/Home';

import {About} from './Components/About';

import {Contact} from './Components/Contact';

function App() {

return (

<div className="container">

<Home/>

<About/>

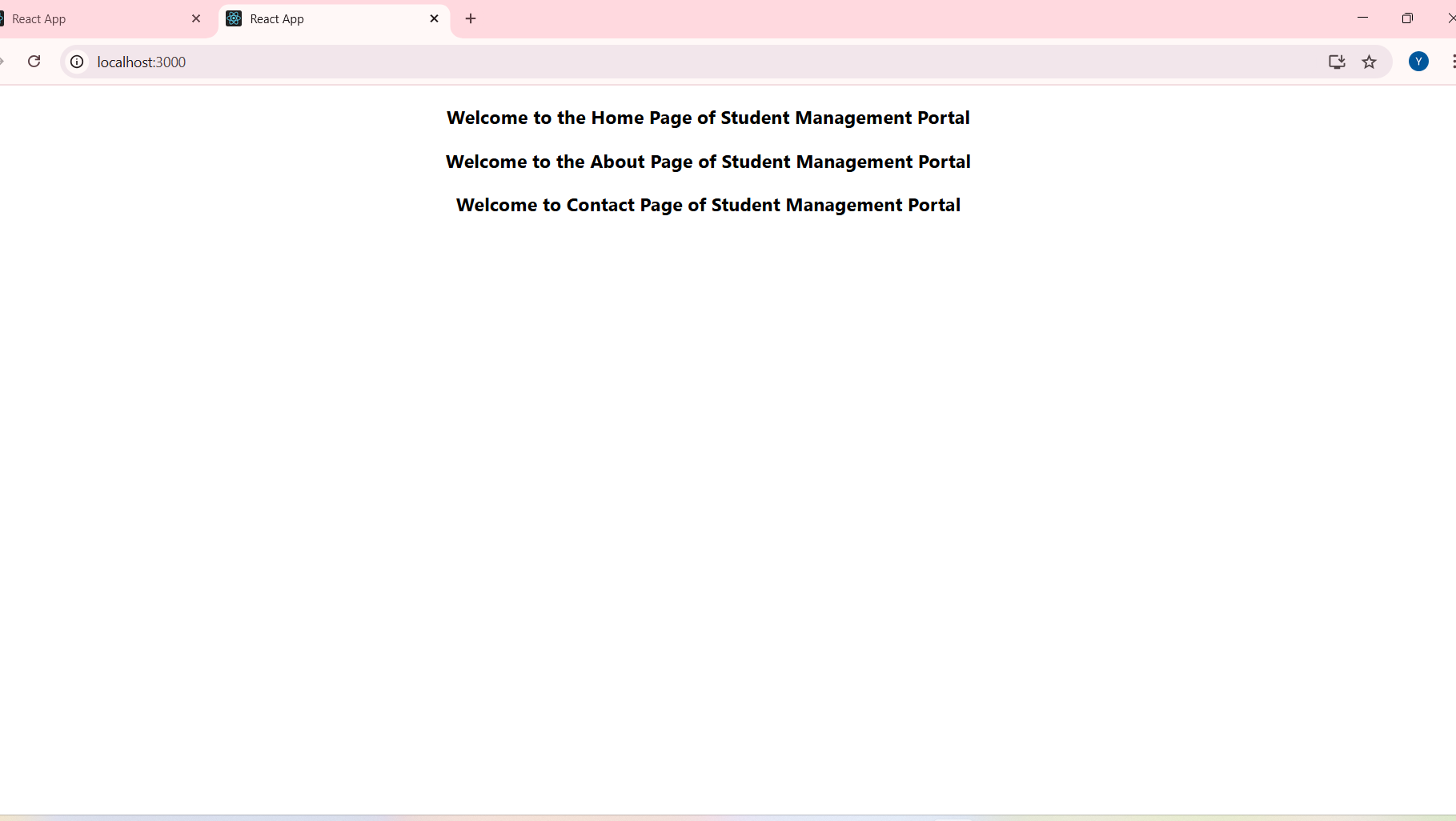
<Contact/>

</div>

);

}

export default App;



**Reactjs HOL 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**

import '../Stylesheets/mystyle.css'

const percentToDecimal=(decimal)=>{

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

}

const calcScore= (total,goal)=>{

return percentToDecimal(total/goal)

}

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

<div className="formatstyle">

<h1><font color="Brown">Student Details:</font></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>Score:</b>

<span>

{calcScore(

total,

goal

)}

</span>

</div>

</div>

);

**Mystyle.css**

.Name{

font-weight: 300;

color:blue;

}

.School{

color:crimson;

}

.Total{

color:darkmagenta;

}

.formatstyle{

text-align: center;

font-size:large;

}

.Score{

color:forestgreen;

}

**App.js**

import logo from './logo.svg';

import './App.css';

import { CalculateScore } from '../src/Components/CalculateScore';

function App()

{

return(

<div>

<CalculateScore Name={"Steeve"}

School={"DNV Public School"}

total={284}

goal={3}

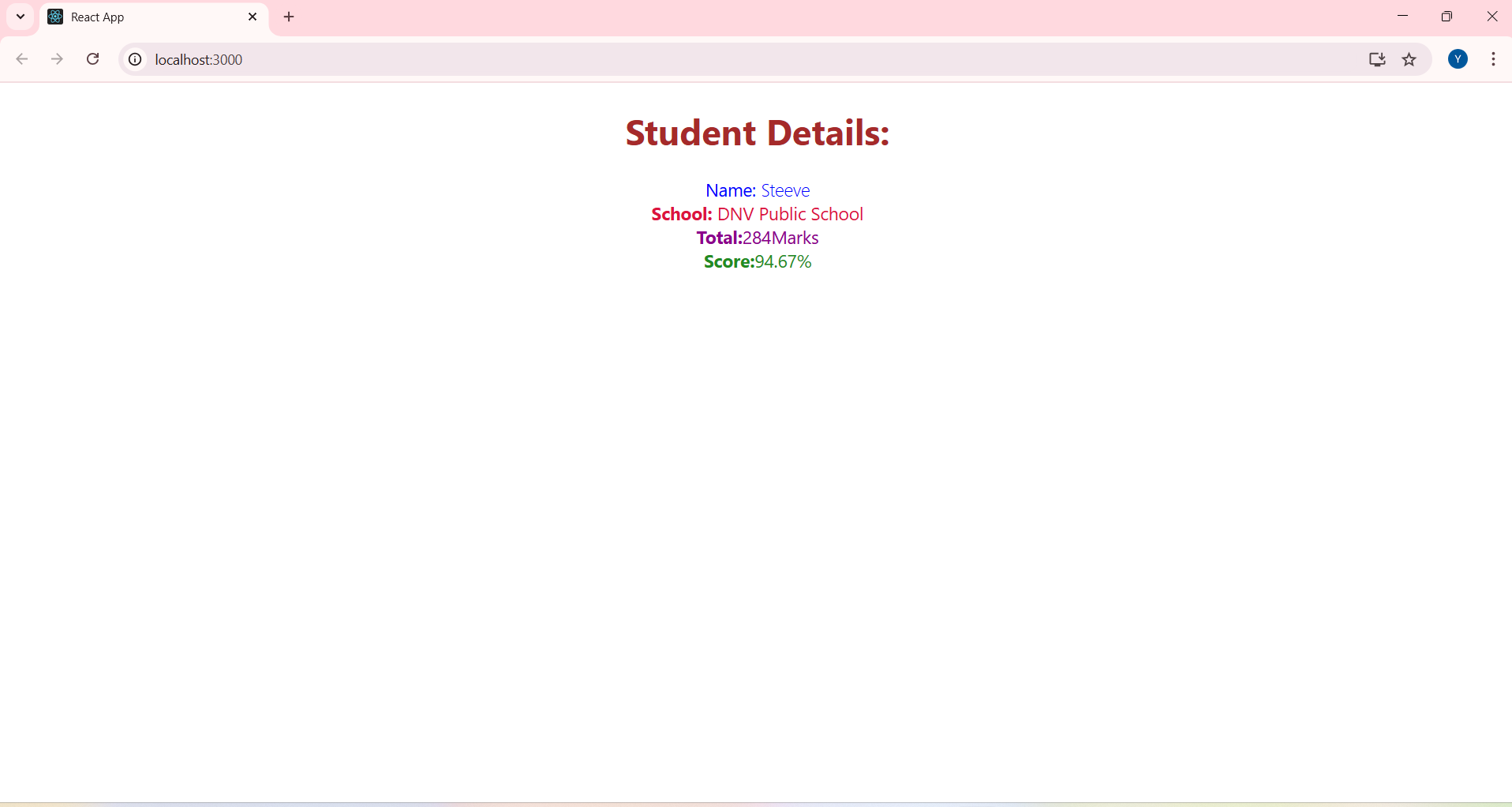
/>

</div>

)

}

export default App;



**React HOL 4**

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

**Post.js**

class Post

{

constructor(id,title,body)

{

this.id=id;

this.title=title;

this.body=body;

}

}

export default Post;

**Posts.js**

import React from "react";

import Post from "./Post";

class Posts extends React.Component {

constructor(props) {

super(props);

this.state = {

posts: [],

hasError: false,

};

}

loadPosts() {

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

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

.then((data) => {

const postObjects = data.map(

(p) => new Post(p.id, p.title, p.body)

);

this.setState({ posts: postObjects });

})

.catch(() => this.setState({ hasError: true }));

}

componentDidMount() {

this.loadPosts();

}

componentDidCatch(error, info) {

alert("An error occurred in Posts component!");

console.error(error, info);

this.setState({ hasError: true });

}

render() {

if (this.state.hasError) return <p>Something went wrong!</p>;

return (

<div>

<h2>Blog Posts</h2>

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

<div key={post.id} style={{ border: "1px solid #ddd", margin: 8, padding: 8 }}>

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

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

</div>

))}

</div>

);

}

}

export default Posts;

**App.js**

import logo from './logo.svg';

import './App.css';

import React from "react";

import Posts from "./Posts";

class App extends React.Component {

render() {

return (

<div>

<h1>blogapp</h1>

<Posts />

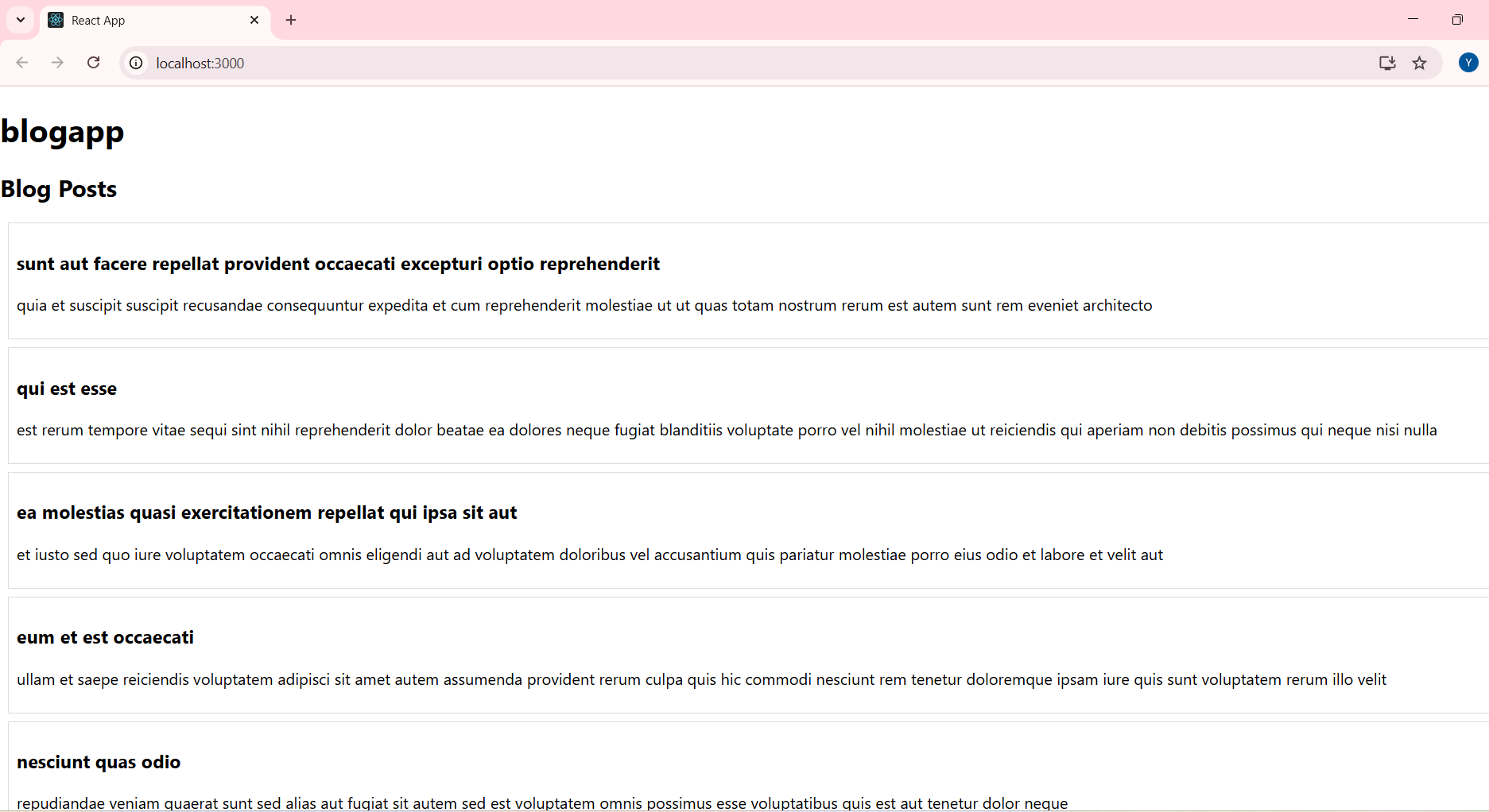
</div>

);

}

}

export default App;



**Reactjs HOL 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.

**CohortDetails.module.css**

.box{

width: 300px;

display: inline-block;

margin: 10px;

padding-top: 10px;

padding-bottom: 10px;

padding-left: 20px;

padding-right: 20px;

border: 1px solid black;

border-radius: 10px;

}

dt{

font-weight: 500;

}

.green{

color:green;

}

.blue{

color:blue;

}

**CohortDetails.js**

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

function CohortDetails(props) {

const isOngoing = props.cohort.currentStatus.toLowerCase() === "ongoing";

return (

<div className={styles.box}>

<h3 className={isOngoing ? styles.green : styles.blue}>

{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>

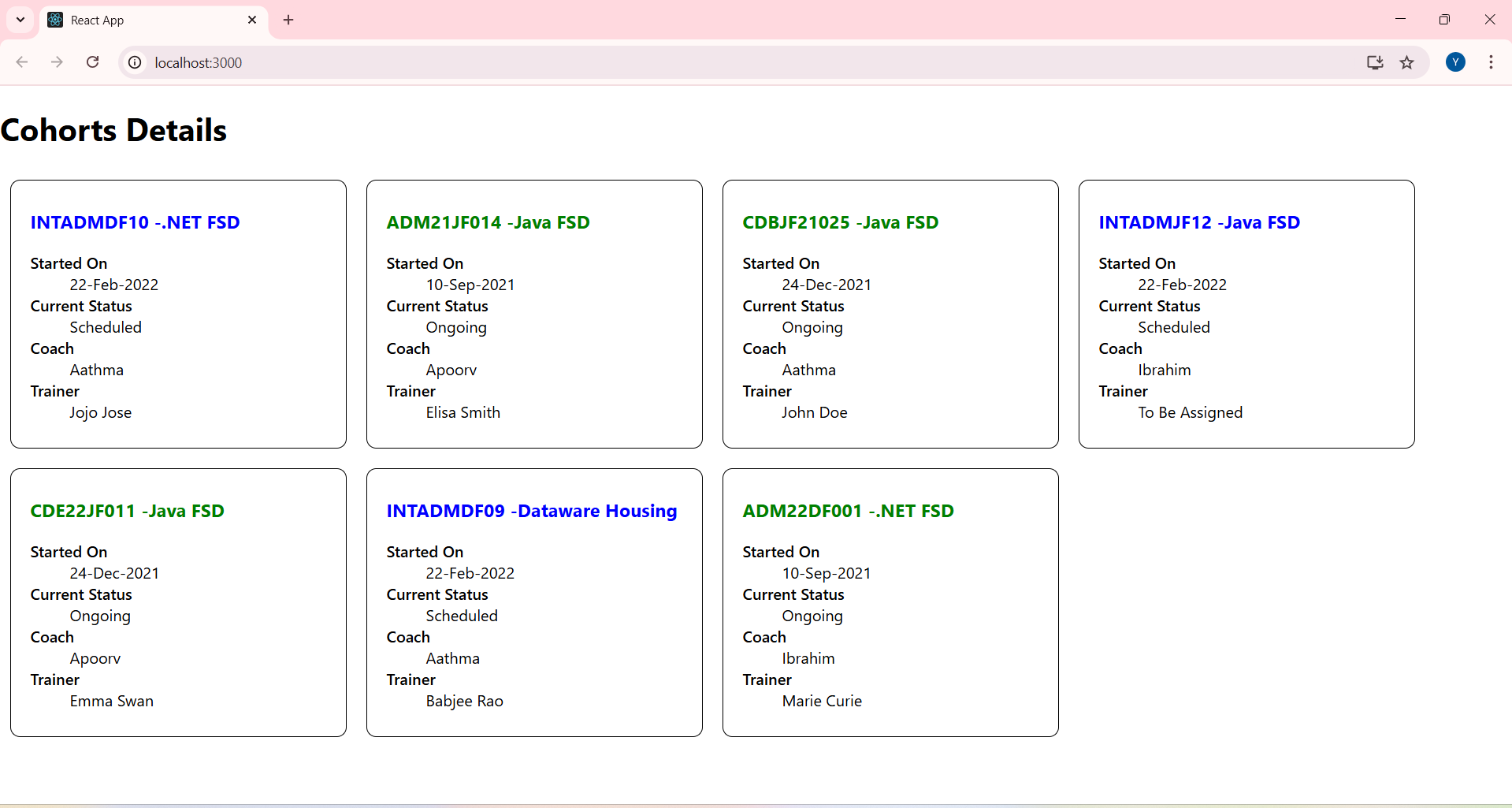
</dl>

</div>

);

}

export default CohortDetails;



**Reactjs HOL 6**

Cognizant Academy teams want to maintain a list of trainers along with their expertise in a SPA using React as the technology. You are assigned the task of creating this React app.

**Trainer.js**

class Trainer {

constructor(trainerId, name, email, phone, technology, skills) {

this.trainerId = trainerId;

this.name = name;

this.email = email;

this.phone = phone;

this.technology = technology;

this.skills = skills;

}

}

export default Trainer;

**TrainerMock.js**

import Trainer from "./Trainer";

const trainers = [

new Trainer(

't-syed8',

'Syed Khaleelullah',

'khaleelullah@cognizant.com',

'97676516962',

'.NET',

['C#','SQL Server','React','.NET Core']

),

new Trainer(

't-jojo',

'Jojo Jose',

'jojo@cognizant.com',

'9897199231',

'Java',

['Java','JSP','Angular','Spring']

),

new Trainer(

't-elisa',

'Elisa Jones',

'elisa@cognizant.com',

'9871212235',

'Python',

['Python','Django','Angular']

)

];

export default trainers;

**TrainersList.js**

import React from 'react';

import { Link } from 'react-router-dom';

const TrainersList = ({ trainers = [] }) => {

console.log("Received in TrainersList:", trainers);

return (

<div>

<h2>Trainers</h2>

<ul>

{trainers.map((trainer, index) => (

<li key={index}>

<Link to={`/trainer/${trainer.trainerId}`}>{trainer.name}</Link>

</li>

))}

</ul>

</div>

);

};

export default TrainersList;

**TrainerDetails.js**

import React from 'react';

import { useParams } from 'react-router-dom';

import trainers from './TrainerMock';

const TrainerDetail = () => {

const { id } = useParams();

const trainer = trainers.find(t => t.trainerId === id);

if (!trainer) {

return <p>Trainer not found</p>;

}

return (

<div>

<h2>{trainer.name} ({trainer.technology})</h2>

<p>{trainer.email}</p>

<p>{trainer.phone}</p>

<ul>

{trainer.skills.map((skill, index) => (

<li key={index}>{skill}</li>

))}

</ul>

</div>

);

};

export default TrainerDetail;

**Home.js**

import React from 'react';

const Home = () => (

<div>

<h1>My Academy Trainers App</h1>

<h2>Welcome to My Academy trainers page</h2>

</div>

);

export default Home;

**App.js**

import React from 'react';

import { BrowserRouter, Routes, Route, Link } from 'react-router-dom';

import Home from './Home';

import TrainersList from './TrainersList';

import TrainerDetail from './TrainerDetails';

import trainers from './TrainerMock';

function App() {

return (

<BrowserRouter>

{}

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

<Link to="/">Home</Link> | <Link to="/trainers">Show Trainers</Link>

</nav>

<Routes>

<Route path="/" element={<Home />} />

<Route path="/trainers" element={<TrainersList trainers={trainers} />} />

<Route path="/trainer/:id" element={<TrainerDetail />} />

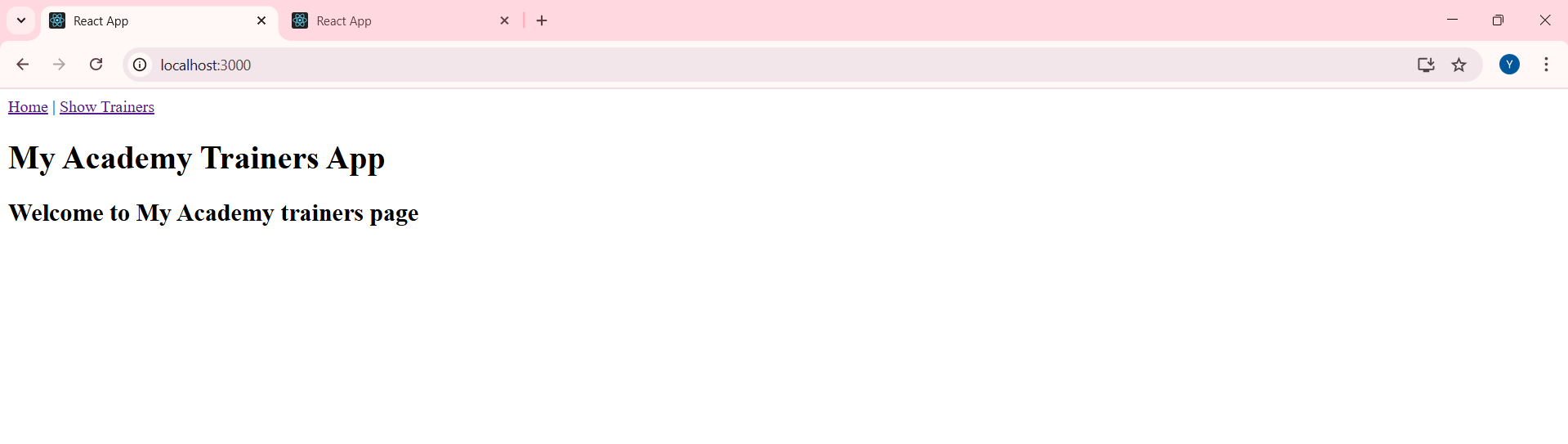
</Routes>

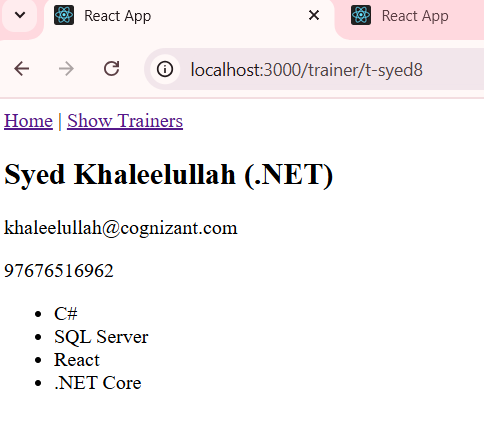
</BrowserRouter>

);

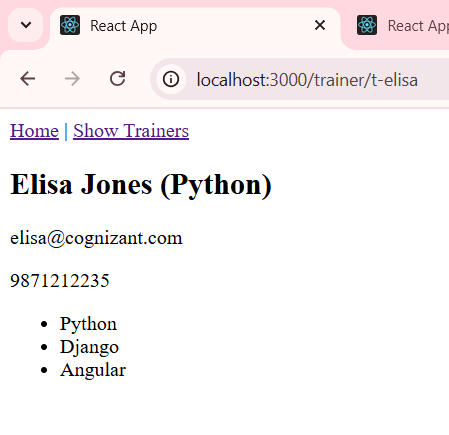
}

export default App;







**Reactjs HOL 7**

Create a React Application named “shoppingapp” with a class component named “OnlineShopping” and “Cart”.

**Cart.js**

import React, { Component } from 'react';

export class Cart extends Component {

render() {

const { itemName, price } = this.props;

return (

<tr>

<td>{itemName}</td>

<td>{price}</td>

</tr>

);

}

}

export default Cart;

**Shoppingcart.js**

import React, { Component } from 'react';

import Cart from './Cart';

import './Styles.css';

export class OnlineShopping extends Component {

render() {

const cartItems = [

{ itemName: 'Laptop', price: 80000 },

{ itemName: 'TV', price: 120000 },

{ itemName: 'Washing Machine', price: 50000 },

{ itemName: 'Mobile', price: 30000 },

{ itemName: 'Fridge', price: 70000 }

];

return (

<div className="container">

<h2 className="heading">Items Ordered :</h2>

<table className="cart-table">

<thead>

<tr>

<th>Name</th>

<th>Price</th>

</tr>

</thead>

<tbody>

{cartItems.map((item, index) => (

<Cart key={index} itemName={item.itemName} price={item.price} />

))}

</tbody>

</table>

</div>

);

}

}

export default OnlineShopping;

**Styles.css**

.container {

text-align: center;

margin-top: 50px;

}

.heading {

color: green;

font-size: 32px;

font-weight: bold;

margin-bottom: 20px;

}

.cart-table {

margin: 0 auto;

width: 260px;

border-collapse: collapse;

border: 1px solid grey;

}

.cart-table th,

.cart-table td {

border: 1px solid grey;

padding: 4px 10px;

background-color: white;

color: black;

font-size: 16px;

text-align: center;

font-weight: normal;

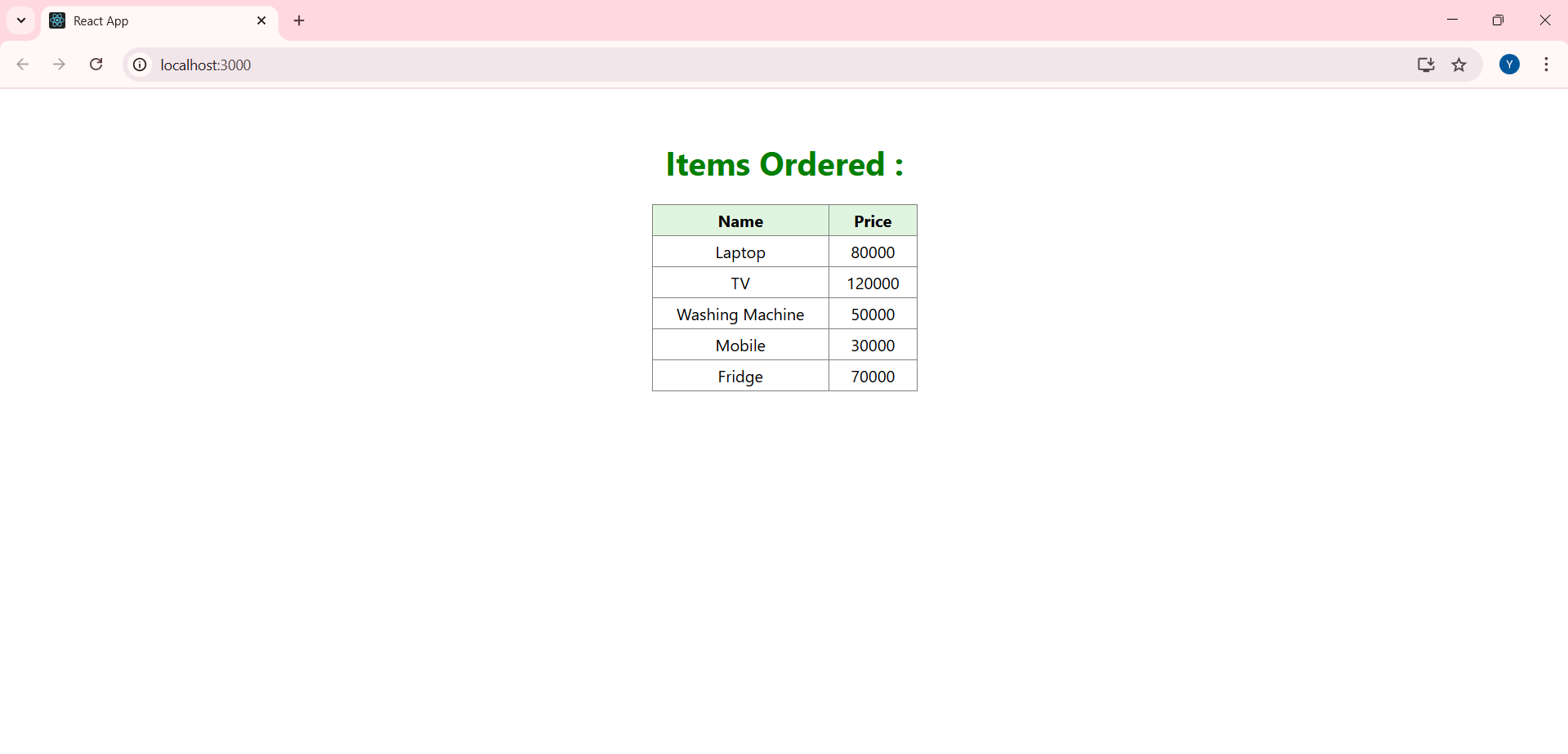
}

.cart-table th {

background-color: #e0f5e0;

font-weight: bold;

}



**Reactjs HOL 8**

Create a React App “counterapp” which will have a component named “CountPeople” which will have 2 methods.

UpdateEntry() 🡪 which will display the number of people who entered the mall.

UpdateExit() 🡪 which will display the number of people who exited the mall.

**App.js**

import React, { Component } from 'react';

import './App.css';

class CountPeople extends Component {

constructor(props) {

super(props);

this.state = {

entryCount: 0,

exitCount: 0,

};

}

updateEntry = () => {

this.setState(prevState => ({

entryCount: prevState.entryCount + 1

}));

}

updateExit = () => {

this.setState(prevState => ({

exitCount: prevState.exitCount + 1

}));

}

render() {

return (

<div className="main">

<div className="section">

<button className="btn" onClick={this.updateEntry}>Login</button>

<span className="text">{this.state.entryCount} People Entered!!!</span>

</div>

<div className="section">

<button className="btn" onClick={this.updateExit}>Exit</button>

<span className="text">{this.state.exitCount} People Left!!!</span>

</div>

</div>

);

}

}

export default CountPeople;

**App.css**

body {

margin: 0;

font-family: -apple-system, BlinkMacSystemFont, 'Segoe UI', 'Roboto', 'Oxygen',

'Ubuntu', 'Cantarell', 'Fira Sans', 'Droid Sans', 'Helvetica Neue',

sans-serif;

-webkit-font-smoothing: antialiased;

-moz-osx-font-smoothing: grayscale;

}

code {

font-family: source-code-pro, Menlo, Monaco, Consolas, 'Courier New',

monospace;

}

