**Week-06 Handson**

1. **Creating my first react app**

* Getting started with setting node on my device.
* Installing create-react-app

*npm install -g create-react-app*

* Creating my react app of name *myfirstapp*

*npx create-react-app myfirstapp*

* Now from cmd we move to the directory of our app, we made

*cd myfirstapp*

* Open the project in VsCode or any Editor then change the content of App functional Component in App.js

*function App() {*

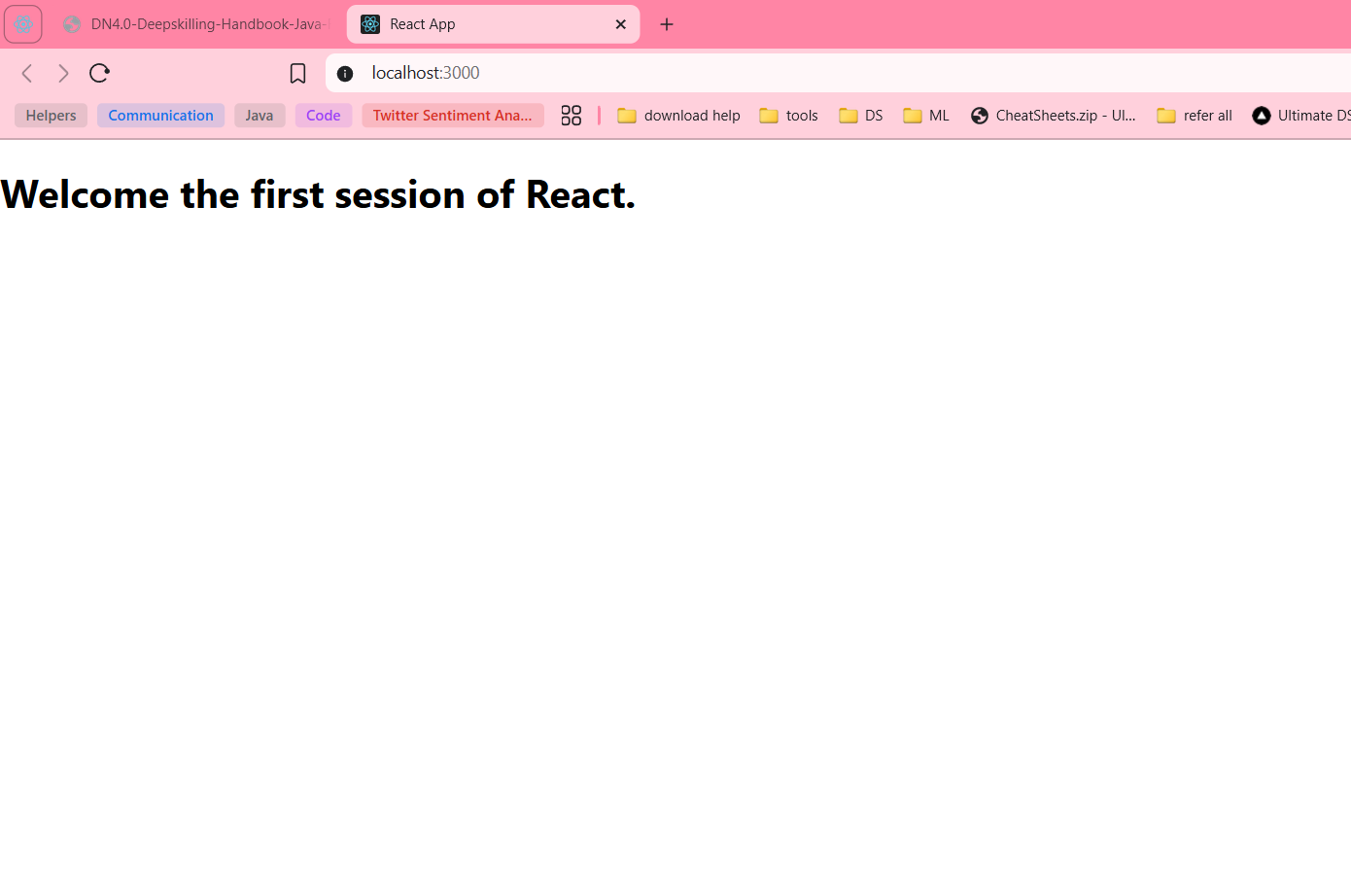
*return (*

*<h1>Welcome the first session of React.</h1>*

*);*

*}*

*export default App;*

**

1. **Creating StudentApp, understanding React Class Component and its rendering**

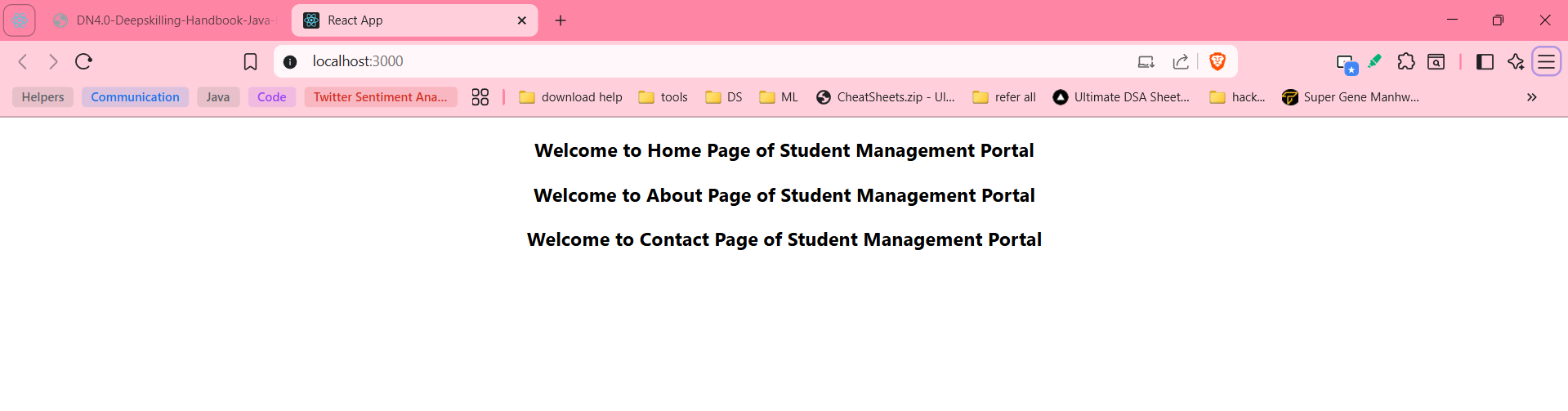
* Create StudentApp
* Under studentapp>src> make a new folder named Components
* Studentapp>src>components> here create a [Home.js, About.js, Contact.js](http://Home.js,About.js,Contact.js)

*Now make this code in all these files*

* import React,{Component} from 'react';
* export class About extends Component {
* render() {
* return(
* <div>
* <h3>Welcome to About Page of Student Management Portal</h3>
* </div>
* );
* }
* }
* Add these class components in the 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;

*start the app*

*npm start*



1. **Creating Score Calculator App, understanding React Functional Component and its rendering**

* Create score-calculator-app

*npm create-react-app*

* Create two folders under src>components, src>Stylesheets
* Under src>Components>

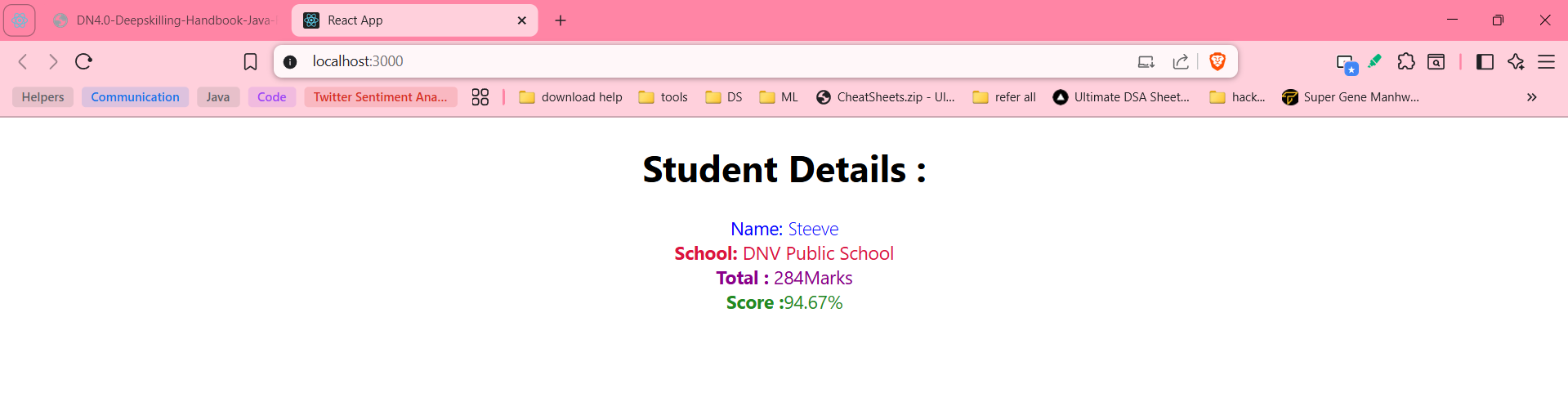
Create CalculateScore.js

* 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 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>Score :</b>
* <span>
* {calcScore(
* total,
* goal
* )}
* </span>
* </div>
* </div>
* )
* *Create mystyle.css under src>Stylesheets*
* .Name
* {
* font-weight:300;
* color:blue;
* }
* .School
* {
* color:crimson;
* }
* .Total
* {
* color:darkmagenta;
* }
* .formatstyle
* {
* text-align:center;
* font-size:large;
* }
* .Score
* {
* color:forestgreen;
* }

***Edit App.js***

* import { CalculateScore } from "./Components/CalculateScore";
* function App(){
* return (
* <div>
* <CalculateScore Name ={"Steeve"}
* School={"DNV Public School"} total={284} goal={3} />
* </div>
* )
* }
* export default App;

Now on starting our app the result is:

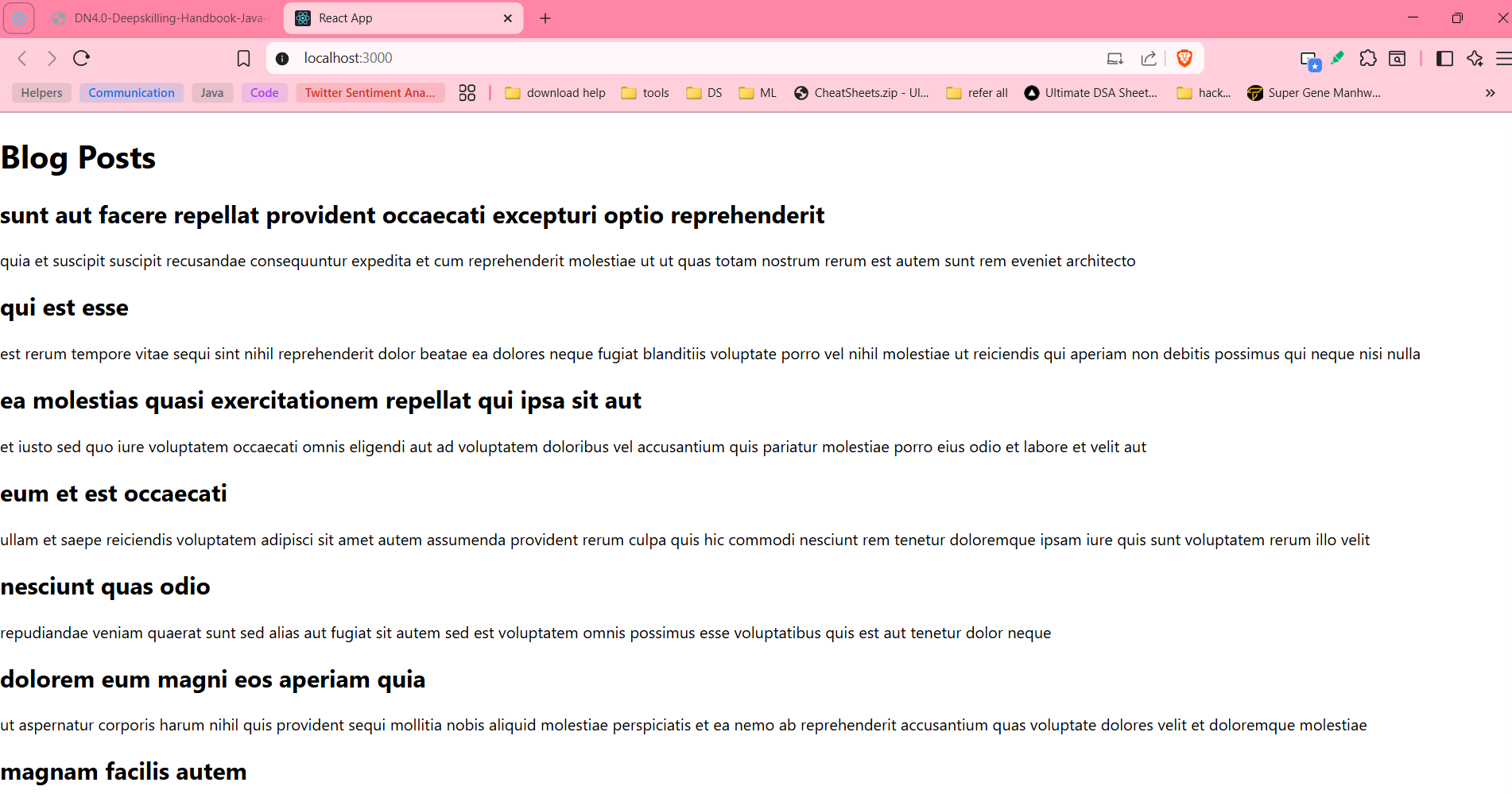


1. **Creating a blogapp to understand ComponentDidMount,ComponentDidCatch**

* Created a Post class Component with a constructor
* class Post{
* constructor(id, title, body){
* this.id = id;
* this.title = title;
* this.body = body;
* }
* }
* export default Post;
* Created a Post class extending React.Component
* It contains

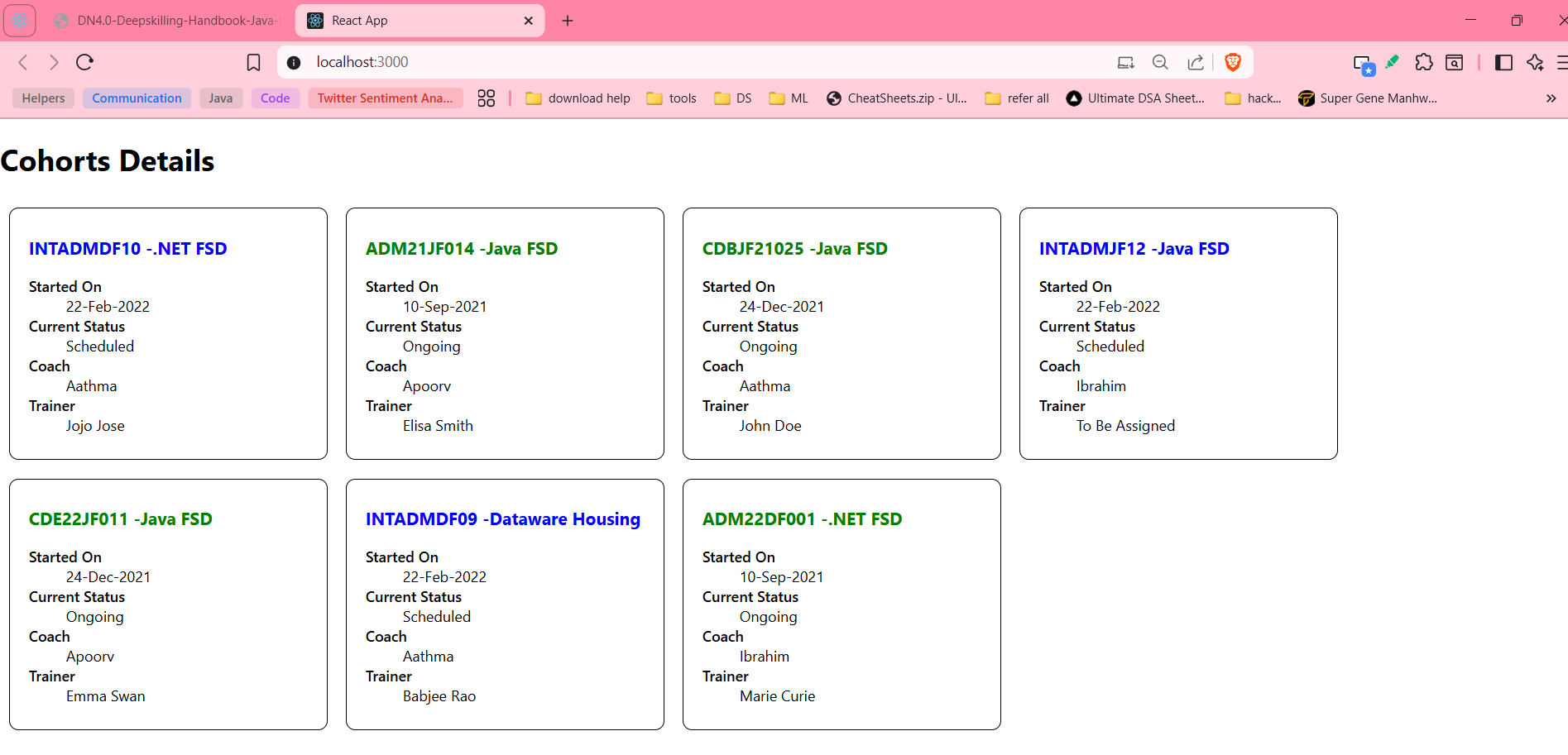
1. a constructor
2. ComponentDidMount
3. ComponentDidCatch
4. loadPosts
5. render

* import React from 'react';
* import Post from './Post';
* class Posts extends React.Component{
* constructor(props){
* super(props);
* this.state = {
* posts: [new Post(1, 'Initial Post 1', 'This is the body of initial post 1'),
* new Post(2, 'Initial Post 2', 'This is the body of initial post 2'),
* ],
* error: false
* };
* }
* loadPosts(){
* return fetch('https://jsonplaceholder.typicode.com/posts')
* .then((res) => res.json())
* .then((data) => this.setState({ posts: data}))
* .catch(() => this.setState({ error: true }));
* };
* componentDidMount(){
* this.loadPosts();
* }
* render(){
* const {error,posts}= this.state;
* if(error){
* return <div><h2 color='red'>Error loading posts.</h2></div>;
* }
* return(
* <div>
* <h1>Blog Posts</h1>
* {posts.map((post) => (
* <div key={post.id}>
* <h2>{post.title}</h2>
* <p>{post.body}</p>
* </div>
* ))}
* </div>
* )
* }
* componentDidCatch(error,info){
* alert("An error occurred: " + error.message);
* }
* }
* export default Posts;
* Added Posts component to App.js
* import logo from './logo.svg';
* import './App.css';
* import Posts from './Posts';
* function App() {
* return (
* <>
* <Posts/>
* </>
* );
* }
* export default App;
* Start the app



1. **Styling a React Component with CSS Module and Inline Styles**

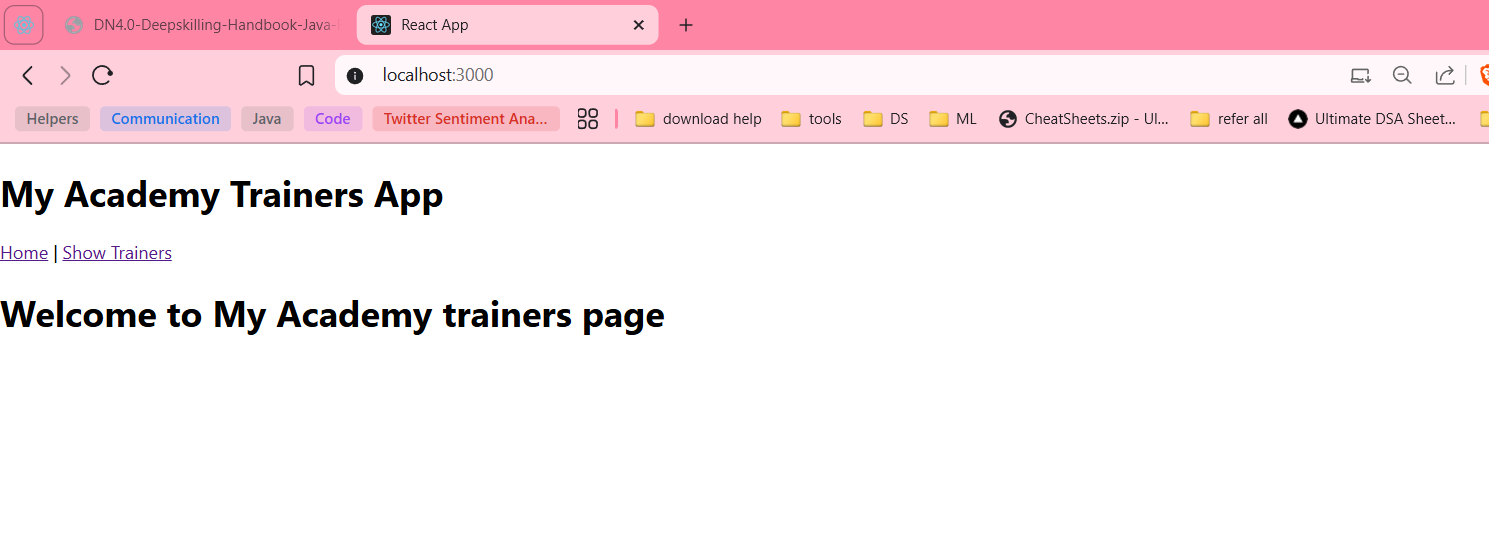
* CohortDetails.js
* import styles from './CohortDetails.module.css';
* function CohortDetails(props) {
* const h3color = {color:props.cohort.currentStatus.toLowerCase()==="ongoing"? "green" : "blue"};
* return (
* <div className={styles.box}>
* <h3 style={h3color}>
* {props.cohort.cohortCode} -
* <span>{props.cohort.technology}</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;
* 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;
* }



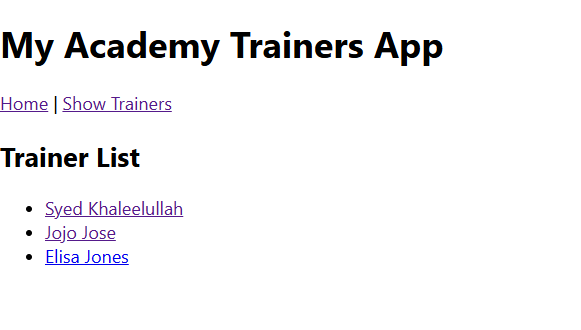
1. **Styling a React Component with CSS Module and Inline Styles**

* Trainer.js
* class Trainer{
* constructor(trainerId,name,eamil,phone,technology,skills){
* this.trainerId = trainerId;
* this.name = name;
* this.email = eamil;
* this.phone = phone;
* this.technology = technology;
* this.skills = skills;
* }
* }
* export default Trainer;
* TrainersMock.js
* import Trainer from './trainer';
* const TrainersMock = [
* 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 TrainersMock;
* TrainerList.js
* import { Link } from 'react-router-dom';
* function TrainerList({ trainers }) {
* return (
* <>
* <h2>Trainer List</h2>
* <ul>
* {trainers.map((trainer) => (
* <li key={trainer.trainerId}>
* <Link to={`/trainers/${trainer.trainerId}`}>{trainer.name}</Link>
* </li>
* ))}
* </ul>
* </>
* );
* }
* export default TrainerList;
* TrainerDetails.js
* import {useParams} from "react-router-dom";
* import trainers from './trainersmock';
* function TrainerDetails() {
* const {id} =useParams();
* console.log("URL param ID:", id);
* const trainer = trainers.find((trainer)=>String(trainer.trainerId)===id);
* if(!trainer){
* return <h2>Trainer not found</h2>;
* }
* return (
* <>
* <h2>Trainer Details</h2>
* <br></br>
* <h3><strong>{trainer.name}({trainer.technology})</strong></h3>
* <br></br>
* <p>{trainer.email}</p>
* <br></br>
* <p>{trainer.phone}</p>
* <br></br>
* <ul>
* {trainer.skills.map((skill)=>(
* <li key={skill}>{skill}</li>
* ))}
* </ul>
* </>
* );
* }
* export default TrainerDetails;
* Home.js
* function Home(){
* return (
* <>
* <h1>
* Welcome to My Academy trainers page
* </h1>
* </>
* );
* }
* export default Home;
* App.js
* import { BrowserRouter as Router, Route, Routes, Link} from 'react-router-dom';
* import Home from './Home';
* import TrainerDetails from './TrainerDetails';
* import TrainerList from './trainerlist';
* import trainers from './trainersmock';
* function App() {
* return (
* <Router>
* <div>
* <h1>My Academy Trainers App</h1>
* <nav>
* <Link to="/">Home</Link> | {" "}
* <Link to="/trainers">Show Trainers</Link>
* </nav>
* <Routes>
* <Route path='/' element={<Home/>}/>
* <Route path="/trainers" element={<TrainerList trainers={trainers}/>}/>
* <Route path="/trainers/:id" element={<TrainerDetails/>}/>
* </Routes>
* </div>
* </Router>
* );
* }
* export default App;

Home page



On clicking Show Trainers



On clicking on a trainer

