Week 6 : React HandsOn

Question 1:

npm install -g create-react-app

npx create-react-app myfirstreact

cd myfirstreact

cd myfirstreact

// src/App.js

import React from 'react';

function App() {

return (

<div>

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

</div>

);

}

export default App;

npm start

OUTPUT : Welcome to the first session of React

Question 2 :

npx create-react-app StudentApp

StudentApp/src/

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;

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;

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;

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>

<Home />

<About />

<Contact />

</div>

);

}

export default App;

Run :

cd StudentApp

npm start

OUTPUT :

Welcome to the Home page of Student Management Portal

Welcome to the About page of the Student Management Portal

Welcome to the Contact page of the Student Management Portal

Question 3 :

npx create-react-app scorecalculatorapp

CalculateScore.js

// src/Components/CalculateScore.js

import React from 'react';

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

function CalculateScore() {

const student = {

name: 'John Doe',

school: 'Cognizant Public School',

totalMarks: 450,

goal: 'Top in class'

};

const average = student.totalMarks / 5;

return (

<div className="container">

<h2>Student Score Calculator</h2>

<p><strong>Name:</strong> {student.name}</p>

<p><strong>School:</strong> {student.school}</p>

<p><strong>Total Marks:</strong> {student.totalMarks}</p>

<p><strong>Goal:</strong> {student.goal}</p>

<p className="average"><strong>Average Score:</strong> {average}</p>

</div>

);

}

export default CalculateScore;

mystyle.css

/\* src/Stylesheets/mystyle.css \*/

.container {

border: 2px solid #4CAF50;

border-radius: 10px;

padding: 20px;

margin: 30px auto;

width: 50%;

background-color: #f0f8ff;

box-shadow: 2px 2px 10px #ccc;

}

h2 {

color: #4CAF50;

text-align: center;

}

.average {

color: darkblue;

font-weight: bold;

font-size: 18px;

}

App.js

// src/App.js

import React from 'react';

import CalculateScore from './Components/CalculateScore';

function App() {

return (

<div className="App">

<CalculateScore />

</div>

);

}

export default App;

OUTPUT :

Student Score Calculator

Name: John Doe

School: Cognizant Public School

Total Marks: 450

Goal: Top in class

Average Score: 90

Question 4:

npx create-react-app blogapp

cd blogapp

code .

Post.js

// src/Post.js

import React from 'react';

class Post extends React.Component {

render() {

return (

<div className="post">

<h3>{this.props.title}</h3>

<p>{this.props.body}</p>

<hr />

</div>

);

}

}

export default Post;

Posts.js

// src/Posts.js

import React from 'react';

import Post from './Post';

class Posts extends React.Component {

constructor(props) {

super(props);

this.state = {

posts: [],

hasError: false

};

}

// Lifecycle method to fetch posts after component mounts

componentDidMount() {

this.loadPosts();

}

// Fetch API to get post data

loadPosts() {

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

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

.then(data => this.setState({ posts: data }))

.catch(error => {

console.error('Error fetching posts:', error);

this.setState({ hasError: true });

});

}

// Lifecycle method to catch rendering errors

componentDidCatch(error, info) {

this.setState({ hasError: true });

alert('Something went wrong in Posts component!');

console.error("Error:", error, "Info:", info);

}

// Render method to display posts

render() {

if (this.state.hasError) {

return <h2>Error loading posts.</h2>;

}

return (

<div>

<h1>Blog Posts</h1>

{this.state.posts.slice(0, 10).map(post => (

<Post key={post.id} title={post.title} body={post.body} />

))}

</div>

);

}

}

export default Posts;

App.js

// src/App.js

import React from 'react';

import Posts from './Posts';

function App() {

return (

<div className="App">

<Posts />

</div>

);

}

export default App;

OUTPUT :

Blog Posts

• sunt aut facere repellat provident occaecati excepturi optio reprehenderit

quia et suscipit...

• qui est esse

est rerum tempore vitae...

• ... and so on (10 posts shown)

Question 5 :

cd path/to/unzipped/react-app

npm install

Cohort.css

/\* src/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;

}

Cohort.js

// src/CohortDetails.js

import React from 'react';

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

function CohortDetails(props) {

const { name, trainer, status } = props;

// Inline style logic

const titleStyle = {

color: status.toLowerCase() === 'ongoing' ? 'green' : 'blue'

};

return (

<div className={styles.box}>

<h3 style={titleStyle}>{name}</h3>

<dl>

<dt>Trainer:</dt>

<dd>{trainer}</dd>

<dt>Status:</dt>

<dd>{status}</dd>

</dl>

</div>

);

}

export default CohortDetails;

App.js

import React from 'react';

import CohortDetails from './CohortDetails';

function App() {

return (

<div className="App">

<h2>My Academy Dashboard</h2>

<CohortDetails name="React Bootcamp" trainer="Anita Sharma" status="Ongoing" />

<CohortDetails name="Java Full Stack" trainer="Rahul Mehra" status="Completed" />

</div>

);

}

export default App;

OUTPUT :

My Academy Dashboard

[ Card 1 ]

React Bootcamp

Trainer: Anita Sharma

Status: Ongoing

[ Card 2 ]

Java Full Stack

Trainer: Rahul Mehra

Status: Completed